

INPUT
Publicación práctica para usuarios de
commodore
Revista mensual 1986
Precio 350 Ptas
Año 2 Número 13
LISTA DE PREMIADOS EN EL SORTEO DE VERANO
DETECCION DE OBJETOS EN PANTALLA
HERRAMIENTAS EN CODIGO MAQUINA
PEEK Y POKE EXPLICADOS
DESENREDA TUS CADENAS



Astérix

Y EL CALDERO MAGICO

EL VIDEO-JUEGO QUE TODOS ESPERABAIS

DISTRIBUIDOR EXCLUSIVO PARA ESPAÑA: ERBE SOFTWARE. C/. STA. ENGRACIA, 17
28010 MADRID. TEL. (91) 447 34 10 - DELEGACION BARCELONA, AVDA. MISTRAL, N.º 10 - TEL. (93) 432 07 31

AÑO 2 NUMERO 13



DIRECTOR:
Alejandro Diges
COORDINADOR EDITORIAL:
Francisco de Molina
DISEÑO GRAFICO:
Tomás López
COLABORADORES:
Luis R. Palencia, Christophe Païs, Francisco Tórtola, Benito Román, Esther de la Cal.
PORTADA:
Foto, Nieves; Maquillaje y peluquería, Grupo Cinco; Modelo, Susana Alguacil.
INPUT Commodore es una publicación juvenil de EDICIONES FORUM
GERENTE DIVISION DE REVISTAS:
Angel Sabat

PUBLICIDAD: Grupo Jota
Madrid: c/ General Varela, 35
Teléf. 270 47 02/03
Barcelona: Avda. de Sarriá, 11-13, 1.º
Teléf. 250 23 99
FOTOMECANICA: Ochoa, S. A.
COMPOSICION: EFCA, S. A.
IMPRESION: Sirven Grafic
C/ Gran Vía, 754-756. 08013 Barcelona
Depósito legal: M. 27.884-1985
SUSCRIPCIONES: EDISA,
López de Hoyos, 141. 28002 Madrid
Teléf. (91) 415 97 12
REDACCION:
Paseo de la Castellana, 93, 14.ª
28046 Madrid. Teléf. 456 54 13
DISTRIBUIDORA
R.B.A. PROMOTORA DE EDICIONES, S. A.
Travesera de Gracia, 56. Edificio Odiseus.
08006 Barcelona.
El precio será el mismo para Canarias que para la Península y en él irá incluida la sobretasa aérea.

INPUT Commodore es una publicación controlada por



INPUT commodore

SUMARIO

EDITORIAL 4

ACTUALIDAD 5

PROGRAMACION
DESENREDA TUS CADENAS 6
SOLUCIONES INGENIOSAS 11
DETECCION DE OBJETOS 40
PUZZLES Y MATEMATICAS 46
COMO HACER REFERENCIAS CRUZADAS 52

APLICACIONES
LA CAJA DE HERRAMIENTAS 17

REVISTA DE SOFTWARE 58

PROGRAMACION DE JUEGOS (COLECCIONABLE) 31
SIMULADOR DE VUELO
DESPEGA PARA TU PRIMER VUELO

NUEVAS TENDENCIAS

Abrumadora ha sido realmente la respuesta al sorteo que propusimos en el número especial de verano. Ahora llega el momento de comprobar si vuestro nombre aparece en la relación de afortunados. Si estáis incluidos, entonces enhorabuena. De lo contrario no desesperéis, es intención de quienes hacemos **INPUT** continuar ofreciendo la posibilidad de obtener algún premio lo más a menudo que permitan las circunstancias. Este es el caso del ganador en el concurso convocado por **Anaya Multimedia** desde la revista. Como resultado acudió a la feria **PCW** en Londres uno de los lectores más habilidosos descifrando claves. Según cuentan las crónicas se divirtió de lo lindo. Septiembre una vez más ha sido pródigo en la aparición de rumores y nuevos productos. La feria británica antes mencionada junto con el **Sonimag** han servido para apreciar cómo parecen confirmarse dos tendencias de cara al futuro de la microinformática. Por un lado los ordenadores destinados a juegos con más memoria, capacidades gráficas y sonido, siendo buena muestra el nuevo **Spectrum+2** y los **MSX 2.** Por otro lado, los ordenadores compatibles **PC** de bajo precio de venta, situándose al alcance de muchos más usuarios, sistemas que hasta hace poco eran un lujo asiático para quienes no fueran empresas o profesionales. El denominador común de ambas tendencias es la bajada de precios aplicada en septiembre. De lo que no cabe duda es que nos encontramos ante el umbral de una nueva etapa.

LOS MEJORES DE INPUT

Hemos pensado que es interesante disponer de un ***ranking*** que ponga en claro, mes a mes, cuáles son los programas preferidos de nuestros lectores. Para ello, es obligado preguntaros directamente y tener así el mejor termómetro para conocer vuestras preferencias. Podéis votar por cualquier programa aunque no haya sido comentado todavía en **INPUT**.

El resultado de las votaciones será publicado en cada número de **INPUT**.

Entre los votantes sortearemos 10 cintas de los títulos que pidáis en vuestros cupones.

Nota: No es preciso que cortéis la revista, una copia hecha a máquina o una simple fotocopia sirven.

Enviad vuestros votos a: **LOS MEJORES DE INPUT** P.º de la Castellana, 93. Planta 14. 28046 Madrid

ELIGE TUS PROGRAMAS

Primer título elegido ____________________

Segundo título elegido ____________________

Tercer título elegido ____________________

Programa que te gustaría conseguir ____________________

Qué ordenador tienes ____________________

Nombre ____________________

1.er Apellido ____________________

2.º Apellido ____________________

Fecha de nacimiento ____________________

Teléfono ____________________

Dirección ____________________

Localidad ____________________

Provincia ____________________

MONSTRUOS POR CORREO

La firma Drosoft acaba de lanzar la versión en castellano del programa "Monstruos por correo", que hace furor entre los usuarios norteamericanos del Commodore 64. En el próximo número de la revista daremos cumplida cuenta de él.

****

RELEVO EN GALERIAS

Thomas Enders, quien desde su cargo como jefe de compras de la división "ON-line" dió el empujón definitivo a la microinformática en los almacenes Galerias Preciados, acaba de regresar a los EE.UU. para ampliar sus estudios en la universidad de Yale.
El distribuidor de software SERMA no lo dudó un momento, haciendole entrega de una placa de agradecimiento por su obra, acto al que acudió una representativa muestra de la prensa especializada.

****

MUSICA Y TECNOLOGIA

Acaba de llegar a nuestras manos el cuarto número de la revista especializada "Música y Tecnología". Su contenido es denso y cuidado, tratando gran variedad de témas que han de interesar a quienes consideran que saber de música no es solo dominar un instrumento.

****

EL 64C EN SONIMAG

Como anunciabamos en estas mismas páginas, Commodore sigue dispuesta a continuar en la linea del 64. El modelo que toma el relevo es el 64C, que ofrece el mismo aspecto externo del 128. La verdadera novedad la constituye el sistema operativo GEOS, que trabaja con ventanas e iconos, haciendo mas fácil el manejo del ordenador frente al teclado. Asi lo pudimos constatar en el reciente Sonimag.

****

NUEVO CATALOGO ABC

La firma ABC analog, una de las pioneras en la importación de software para microordenadores, nos acaba de remitir su catalogo del 15 de septirmbre. En el continua con su fórmula de aglutinar los titulos de juego de mayor éxito del momento. cabe resaltar que en la sección del Commodore 16 aparecen 17 referencias.

****

NUEVA LINEA INVES

Investrónica, la empresa que implantó los ordenadores de Sinclair en nuestro país, acaba de optar por una nueva estrategia en su intento de poner la electrónica de consumo al alcance de todos.
Por un lado entra en los equipos de sonido con una cadena llamada 100 HF, que puede ver sus prestaciones ampliadas en base a un disco compacto y el sintonizador digital en el modelo CD 300 HF, aunque tambièn existe el compact disk por separado, DISC CD 200.
Por otro lado entra en el mundo de los ordenadores compatibles con el estándar PC con tres modelos: INVES 256X, PC 640A y PC 640X.
El denominador común de toda la línea INVES es su bajo precio de venta en realación con sus prestaciones.
La entrada de estas grandes firmas distribuidoras parece que vienen a hacer realidad el gran sueño de sir Clive Sinclair, quien hace tiempo lanzó equipos de HIFI a bajo precio en el Reino Unido, aunque utilizando la tecnología mas limitada de aquel momento.

****

DESENREDA TUS CADENAS

- COMPARACION Y ORDENACIO DE CADENA
- TROCEAD
- ¿COMO ES DE LARGA LA CADENA
- EL PROCESO DE TEXTO

Las cadenas de caracteres se utilizan en toda clase de programas; de hecho se usan en todas las aplicaciones que requieran algo más que simples números. Aquí presentamos unas cuantas maneras de sacar de ellas el máximo provecho.

Una cadena está constituída por una colección de caracteres. Puede tratarse de letras, números, signos de puntuación o cualquier otro de los símbolos del teclado. Además puedes ponerlas juntas en el orden que quieras.

Naturalmente lo normal es que una cadena contenga algún tipo de información útil. Por ejemplo, la cadena «PEDRO DIAZ 241061 S» contiene el nombre, el primer apellido, la fecha de nacimiento y el estado civil de una determinada persona. En total cuatro elementos de información. A su vez la fecha de nacimiento se puede subdividir en día, mes y año, por lo que en realidad hay un total de seis elementos de información.



Son muchas las ocasiones en las qu suele ser necesario subdividir (tro cear) una cadena de caracteres par extraer determinadas porciones de in formación, tal como ocurre con la fe cha del ejemplo anterior. En otra ocasiones puede ser necesario suma las cadenas. Puede que a veces tam bién quieras medir la longitud de la ca dena o calcular el valor de alguna d sus partes numéricas. Todo esto s puede hacer utilizando unas cuanta **palabras clave** del BASIC.

La suma de cadenas, la llamad concatenación, es la más fácil de esta operaciones. Para ejecutarla no tiene más que utilizar el símbolo +. Si po ejemplo A$ es igual a «HE» y B$ e igual a «AQUI», entonces A$ + B es igual a «HE AQUI». La concate nación es una operación que se limit a juntar unas cadenas de caractere con otras; debe quedar claro que pon unas a continuación de otras, pero n suma sus valores numéricos. Asi «439» + «241» es igual a «439241» no a 680.

COMPARACION DE CADENAS

El ordenador, además de pone unas cadenas junto a otras, es capa de compararlas para ver si son igua les, como ocurre en este juego de adi vinanzas:

Teclea para C-64 y Vic-20

```
10 G=1:ON INT(RND(1)*6)+1
   GOTO 20,30,40,50,60,70
20 B$="MANZANA":GOTO 80
30 B$="NARANJA":GOTO 80
40 B$="PLATANO":GOTO 80
50 B$="LIMON":GOTO 80
60 B$="MELON":GOTO 80
70 B$="PERA":GOTO 80
80 PRINT"[SHIFT+CLR/HOME]SO
   UNA FRUTA, CUAL SOY?"
90 INPUT A$
```

```
100 IF A$=B$ THEN GOTO 155
110 G=G+1
120 PRINT"MAL!"
130 FOR J=1 TO 2000
140 NEXT J
150 GOTO 90
155 IF G=1 THEN PRINT "ACER
    TASTE A LA PRIMERA":END
160 PRINT "ACERTASTE TRAS";G
    X;"INTENTOS"
170 END
```

La línea 10 pone a 1 el contador de adivinanzas y a continuación lanza un dado electrónico para elegir una fruta. Las diferentes opciones están almacenadas en las líneas 20 a 70. Cualquiera que sea la línea a la que vaya el ordenador, el nombre de la fruta correspondiente se almacena en B$, pasando después a la línea 80. Ahora tienes que acertar cuál es la fruta elegida e introducir tu conjetura mediante una sentencia INPUT con la variable A$. Esta variable se compara con A$ y si hay coincidencia total entre ambas, el ordenador presenta el mensaje «HAS ACERTADO UNA VEZ» o bien «HAS ACERTADO» tantas «VECES» como corresponda en cada caso. Si A$ no es igual a B$, el ordenador escribe «ERROR» y tienes que ensayar de nuevo una respuesta. El programa sigue adelante hasta que la respuesta sea la correcta.

Fíjate bien en que aunque la respuesta elegida por tí sea la correcta, la condición A$ = B$ no se cumplirá a menos que la ortografía sea también correcta. Para que sea correcta, las dos cadenas de caracteres han de ser absolutamente idénticas, tanto las letras, como los números y los signos de puntuación.

Puedes utilizar esta técnica de comparación de cadenas de caracteres para comprobar las entradas introducidas desde el teclado, con una línea tal como la siguiente:

```
IF A$="SI" THEN PRINT "ESTAS
SEGURO?"
```

Observa que la condición anterior no se satisface si la palabra «sí» se teclea con letras minúsculas.

ORDENACION DE CADENAS

También puedes comparar las cadenas de caracteres utilizando los signos de desigualdad < y >. Podrías usarlos en una línea como la siguiente:

```
IF A$<B$ THEN PRINT "LA PRI
MERA ES ";A$
```

En este caso en la condición A$ < B$ se examina si la cadena A$ se sitúa antes o después que B$ cuando ambas se colocan en orden alfabético. Pero, ¡cuidado!, el ordenador hace la ordenación alfabética atendiendo al código ASCII de cada letra: a la A le corresponde el código ASCII 65 y a la Z el 90. El problema es que las letras minúsculas también tienen sus códigos ASCII: a la a le corresponde el 97 y a la z el 122. De esta forma, todas las cadenas de caracteres que empiezan por letras mayúsculas se sitúan por delante de las otras.

Y lo que es peor, los números, los signos de puntuación, los espacios y demás signos, también tienen sus códigos ASCII, por lo que el orden resultante para las cadenas puede quedar totalmente trastocado. No obstante, utilizando cuidadosamente los signos < y > se puede obtener una ordenación alfabética de las cadenas de caracteres.

TROCEADO DE CADENAS

Se puede extraer un carácter o un conjunto de caracteres del interior de una cadena. Esto se hace utilizando las funciones LEFT$, RIGHT$ y MID$.

LEFT$(A$,numero) empieza por el principio o extremo izquierdo de la cadena A$ y te da el número de caracteres que especifiques. Si la cadena A$ es «SR JUAN PEREZ» y especificas dos caracteres, es decir, si tecleas LEFT$(A$,2), el resultado será «SR».

Análogamente la función RIGHT$ empieza a contar por el otro extremo de la cadena. Así, RIGHT$(A$,5) dará como resultado «PEREZ».

Con la función MID$ puedes especificar dos números, que corresponden a la posición de partida y al número de caracteres que ha de contener el trozo. Por ejemplo, MID$(A$,4,6) empezará en el cuarto carácter J y tomará seis caracteres, dando como resultado «JUAN P». Si solamente especificas un número, tal como MID$(A$,4), obtendrás todos los caracteres desde el cuarto en adelante.



El siguiente programa utiliza las funciones LEFT$, MID$ y LEN. Se trata de un juego de anagramas para dos personas. Una de ellas introduce una palabra que el ordenador seguidamente cifra y presenta en forma de anagrama para que la otra persona la adivine.

```
10 PRINT "[SHIFT+CLR/HOME]"
20 PRINT"PROGRAMA DE LA
   ADIVINANZA"
30 PRINT"ESCRIBE LA PALABRA
   A ENCONTRAR"
```



```
50 INPUT">[CTRL+7]";W$
55 WO$=W$
70 PRINT "[SHIFT+CLR/HOME]
    [COMM.+7]"
80 FOR N=LEN(W$) TO 1 STEP-1
90 M=INT(RND(1)*N)+1
100 A$=A$+MID$(W$,M,1)
110 W$=LEFT$(W$,M-1)+RIGHT$
    (W$,LEN(W$)-M)
120 NEXT N
130 PRINT "EL ANAGRAMA ES ";
    A$
140 PRINT "CUAL PIENSAS QUE
    ES LA PALABRA?"
160 INPUT GU$
170 G=G+1
180 IF GU$<>WO$ THEN PRINT
    "ERRONEA, INTENTA OTRA
    VEZ":GOTO 160
190 PRINT "BIEN HECHO"
195 IF G=1 THEN PRINT "NECE
    SITASTE 1 INTENTO":GOTO
    210
200 PRINT "NECESITASTE";G;
    "INTENTOS"
210 PRINT"OTRA PARTIDA
    (S/N)?"
220 GET A$:IF A$<>"S" AND A$
    <>"N" THEN 220
230 IF A$="S" THEN RUN
```

```
50 INPUT ">[CTRL+2]";W$
70 PRINT "[SHIFT+CLR/HOME]
   [CTRL+7]"
```

Para el VIC-20 introduce los siguientes cambios:

Cuando ejecutes este juego, observarás que la primera palabra tecleada no aparece en la pantalla. Lo hemos programado así para que tu oponente no pueda saber cuál es. Cada ordenador utiliza un método diferente para conseguir esto. El **Commodore** lo que hace es escribir la palabra con el color de fondo de pantalla, de manera que resulta invisible.

La rutina de cifrado está contenida en las líneas 80 a 120. Lo que hace esta rutina es extraer caracteres de forma aleatoria de la pantalla y ponerlos todos juntos en A$ que se convierte así en un anagrama.

En la línea 90 se elige un número aleatorio M comprendido entre 1 y la longitud de la palabra, y a continuación la línea 100 extrae la letra M-sima y se la añade a A$. La línea 110 borra este carácter de la palabra original, suprimiendo su parte izquierda hasta el número M y añadiendo la parte restante que queda a la derecha de M. La palabra resultante tiene ahora un carácter menos, pero la vez siguiente, la variable N también es un carácter menor, por lo que el número aleatorio M está otra vez restringido a la longitud de palabra.

Llegará un momento en que se hayan extraído todos los caracteres y el anagrama quedará impreso en la pantalla, momento en que el programa le preguntará a tu oponente cuál era la palabra original. Cuando teclees en la sentencia INPUT la palabra correcta, el programa te responde cuántos intentos has necesitado y le brinda una nueva oportunidad.

Sería muy sencillo modificar este programa para que se puedan leer las palabras a través de una serie de sentencias DATA en vez de utilizar separadamente un INPUT cada vez. También puedes incluir un sistema de puntuaciones que, por ejemplo, asigne diez puntos cada vez que se acierte a la primera, nueve puntos a la segunda y así sucesivamente.

Otro posible uso de este método de troceado es la manipulación de fechas. Incluso en los casos en que las fechas se introducen en forma de números, por ejemplo 27/03/51, en vez del 27 de marzo de 1951. Después de todo, puedes manipular una fecha expresada de esta forma con las leyes normales de las matemáticas. También puedes procesar diferentes partes de una fecha por medio de la aritmética. Puedes calcular por ejemplo la edad de una determinada persona tecleando su fecha de nacimiento, o calcular el número de días transcurridos entre dos fechas dadas; no tienes más que manejar separadamente la parte de los años, la de los meses y la de los días.

Las funciones LEFT$, RIGHT$ y MID$, te permiten separar con facilidad la parte de los días, la de los meses y la de los años de una determinada fecha. Si ahora utilizas las funciones VAL, obtendrás a partir de la cadena de caracteres resultante un número con el que puedes hacer sumas, restas multiplicaciones y divisiones.

LONGITUD DE LAS CADENAS

A veces interesa conocer la longitud de una cadena. Si por ejemplo sólo dispones de una limitada capacidad de memoria para cada elemento de información introducido desde el teclado, o sólo un espacio limitado de pantalla para presentar dicha información, puede ser de gran ayuda comprobar la longitud de la cadena antes de seguir con el programa.

La función LEN(A$) te da como resultado el número de caracteres que contiene la cadena A$. Se trata de una función numérica y no de una cadena

de caracteres, por lo que puedes manipularla con arreglo a las leyes normales del álgebra. Por ejemplo, si A$ = «Sr Juan Perez», LEN(A$) = 13. Pero supongamos que te piden que teclees un nombre para un determinado fichero cuyo formato tabulado para representación en pantalla sólo tiene sitio para 11 letras, por lo que puede ser que haya que enviar un mensaje para que se recorte la entrada, tal como el siguiente:

Teclea para C-64 y Vic-20

```
10 PRINT "ESCRIBE EL NOMBRE"
20 INPUT A$
30 IF LEN(A$)>11 THEN PRINT
   "LO SIENTO. SOLO ADMITO
   11 CARACTERES":GOTO 10
```

El usuario podría así recortar su entrada y teclear únicamente «JUAN PEREZ».

En otros casos podría resultar más fácil truncar automáticamente la entrada. Esto se hace con una línea tal como la siguiente:

```
IF LEN (A$)>15 THEN
A($)=LEFT$(A$,15)
```

CONVERSION DE CADENAS A NUMEROS

Uno de los usos de las variables de cadenas de caracteres es conseguir que las entradas introducidas por el teclado sean correctas. Por ejemplo, si escribes un programa que contenga un par de líneas como las siguientes:

```
100 PRINT "ESCRIBE UN
    NUMERO"
110 INPUT A
```

el ordenador se quedará esperando que teclees algún número. Si por error tecleas algún carácter no numérico, casi todos los ordenadores interrumpirán la representación en pantalla y enviarán un mensaje estándar de error. Dicho mensaje te informará de que hay algo que está mal pero no te dirá lo que es.

Pero si en lugar de esto utilizas:

```
110 INPUT A$
```

en este caso el usuario puede teclear casi cualquier cosa y el ordenador lo aceptará. El siguiente paso es convertir la cadena de caracteres en un número. Para hacerlo, utiliza la función VAL(A$).

La función VAL en los **Commodore** extrae la parte numérica de la cadena de caracteres. Así, si A$ es un número, entonces VAL(A$) será un número que puede ser utilizado en el resto del programa. Si por el contrario A$ no es un número, VAL(A$) dará como resultado el valor 0. Puedes escribir entonces una pequeña subrutina que explique en detalle lo que el usuario ha hecho mal y le brinde otra oportunidad sin tener que salir del programa y destruir la presentación que haya en pantalla.

Un punto importante a tener en cuenta a propósito de la función VAL es que sólo extrae números del principio de la cadena. En consecuencia, VAL(«25 DE JULIO») será igual a 25, mientras que VAL(JULIO 25) será igual a cero. De modo que ten cuidado al utilizarla.

La función VAL resulta muy útil para ordenar los números de una cadena de caracteres de varias formas diferentes. Si por alguna razón tienes los nombres y las puntuaciones de un grupo de alumnos en forma de cadenas, tal como: A$ = «32 JORGE», B$ = «45 MIGUEL», C$ = «41 ELENA», etc y deseas calcular la media de la clase, puedes extraer valores individuales con ayuda de la función VAL. Así VAL(A$) vale 32, VAL(B$) vale 45 y VAL(C$) vale 41.

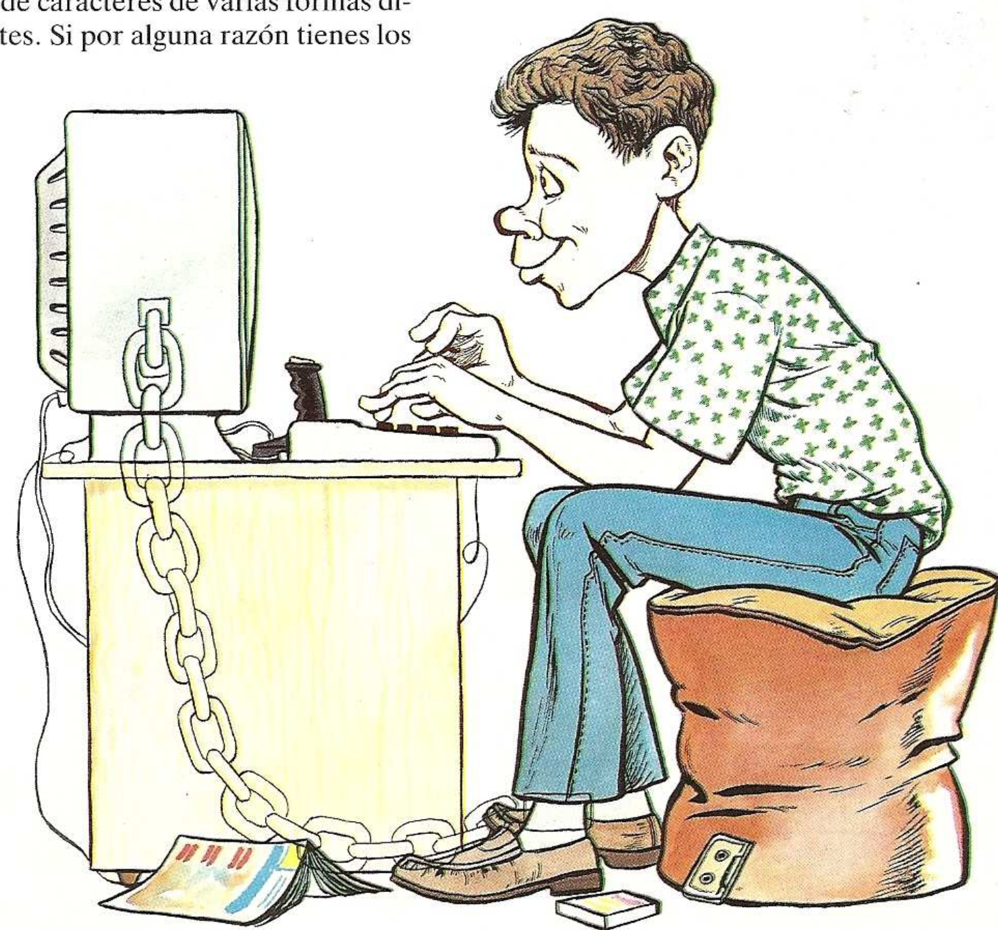
También puedes usar VAL de forma parecida para ignorar las unidades que pueden ir acompañando magnitudes tales como pesos y longitudes. El siguiente programa te permitirá apreciar la diferencia:

Teclea para C-64 y Vic-20

```
100 A$="32 KG"
110 B$="110 KG"
120 PRINT "A$+B$=";A$+B$
130 PRINT "VAL(A$)+VAL(B$)="
    ;VAL(A$)+VAL(B$)
140 END
```

CONVERSION DE NUMEROS A CADENAS

La función STR$ tiene un resultado casi opuesto al de VAL. Sirve para



ORDENA TUS CADENAS

Cuando el ordenador extrae de manera casi mágica las direcciones de centenares de entradas separadas y las presenta en la pantalla, es fácil imaginar que la máquina es más inteligente de lo que realmente es. Sucede casi como si el ordenador estuviera leyendo directamente las entradas y a continuación decidiendo lo que tiene que hacer.

No te confundas. El ordenador se limita a extraer una parte de la cadena a medida que va recibiendo información. Si resulta que una parte de esas cadenas contiene un galimatías, pues eso es lo que te encontrarás.

Para asegurarte de que obtienes la misma información de cada cadena, tienes que garantizar que la información se almacena siempre en el mismo sitio. Los investigadores profesionales con frecuencia utilizan tarjetas de entrada con un formato normalizado, que constituyen una de las ayudas más valiosas que te puedes procurar. Para ello no necesitas más que una serie de espacios que contengan los caracteres, rotulados con arreglo al lugar donde debe empezar y acabar cada elemento de información.

transformar una cadena en un número. La ventaja de esto es que las cadenas de caracteres pueden ser manipuladas de forma distinta a los números, por ejemplo con ayuda del troceado y la concatenación que ya hemos visto, por lo que STR$ tiene varias aplicaciones.



El siguiente programa sirve para convertir en binario un número decimal. Aunque los ordenadores manejen toda su aritmética interna en binario, el lenguaje BASIC sólo puede manejar números decimales y, en algunos casos, hexadecimales. Por ello, cuando en un programa en BASIC se presenta un número en binario, hay que manejarlo como una cadena de caracteres.

Teclea para C-64 y Vic-20

```
10 PRINT "DECIMAL A BINARIO"
20 PRINT "ESCRIBE UN ENTERO
   DECIMAL"
30 INPUT D
40 B$=""
50 B$=STR$(D-INT(D/2)*2)+B$
60 D=INT(D/2)
70 IF D<>0 THEN GOTO 50
80 PRINT "EL NUMERO BINARIO
   ES ";B$
```

Al introducir un número decimal positivo, la línea 40 hace la cadena B$ igual a la cadena nula, que después va llenando progresivamente con dígitos a medida que progresa el cálculo en el bucle de las líneas 50 a 80.

La línea 50 es la que realmente se encarga de construir el número binario. Para ello resta dos veces la parte entera de la mitad del número decimal del propio número decimal. Esta es una forma sencilla de comprobar si un número es par o impar. Si es impar, el resultado es 1, mientras que siendo par el resultado es 0. Naturalmente, estos son los dígitos binarios buscados.

PROCESAMIENTO DE TEXTOS

Las funciones de manejo de cadenas de caracteres se utilizan ampliamente, en especial en proceso de textos. Una necesidad que se plantea con frecuencia es la sustitución de una palabra por otra en toda la extensión de un documento (posiblemente debido a que has descubierto una falta de ortografía). Naturalmente, si la nueva palabra tiene una longitud diferente de la original, tienes que desplazar el resto del texto para dejarle sitio. El siguiente programa te muestra la manera de hacerlo:

Teclea para C-64 y Vic-20

```
10 INPUT "ESCRIBE EL TEXTO"
   ;T$
20 INPUT "PALABRA A REEMPLA
   ZAR";P$
30 INPUT "PALABRA SUSTITUTA"
   ;NP$
35 P=0
40 P=P+1
50 A$=MID$(T$,P,LEN(P$))
60 IF A$<>P$ THEN 90
70 T$=LEFT$(T$,P-1)+NP$+
   RIGHT$(T$,LEN(T$)-P-LEN
   (P$)+1)
80 P=P+LEN(NP$)-1
90 IF P<LEN(T$) THEN
   GOTO 40
100 PRINT T$
110 GOTO 20
```

Empieza introduciendo una frase más bien sencilla y a continuación ensaya el efecto de sustituir algunas de sus letras o palabras. En las palabras cortas tales como «a» o «un» conviene teclear además un espacio antes y después para evitar que se apliquen también los cambios cada vez que se presenten dichas palabras embebidas en otras más largas, tales como «para» o «unidad». La palabra buscada ha de ir entre comillas: «un».

El programa funciona de la siguiente manera: La línea 40 envía a P a empezar la fase de búsqueda al principio del texto. Las líneas 50 y 60 detectan la primera aparición de la palabra o grupo de letras que pretendes sustituir y a continuación la línea 70 realiza la sustitución. Lo que hace es tomar el texto original, remplazar la palabra antigua por la nueva y añadir el texto restante. El proceso se repite hasta que se ha realizado la sustitución en todos los sitios en que aparece la palabra que se quiere sustituir, imprimiéndose a continuación en la línea 100.

Como has visto, se trata de un ejemplo muy simple, los procesadores del mundo real son mucho más complicados. Sin embargo puede servir para darte una idea de algunas aplicaciones prácticas de lo que se puede hacer con las cadenas de caracteres.

SOLUCIONES «INGENIOSAS»

- LA MAQUINA MAS SENCILLA
- VENTAJAS MECANICAS
- ALCANZANDO MAYORES ALTURAS
- AHORRO DE ESFUERZOS
- EL ARIETE HIDRAULICO

La mecánica no existe sólo para los ingenieros. Es el estudio de las interacciones entre las fuerzas e interviene en todo tipo de actividades cotidianas, susceptibles de ser analizadas por tu ordenador.

La construcción de muchos de los mayores monumentos del mundo, tales como las pirámides de Egipto o el complejo megalítico de Stonehenge, resulta especialmente sorprendente por el hecho de que fueron construídos mucho tiempo antes del desarrollo de los dispositivos tecnológicos considerados como necesarios para dichas tareas.

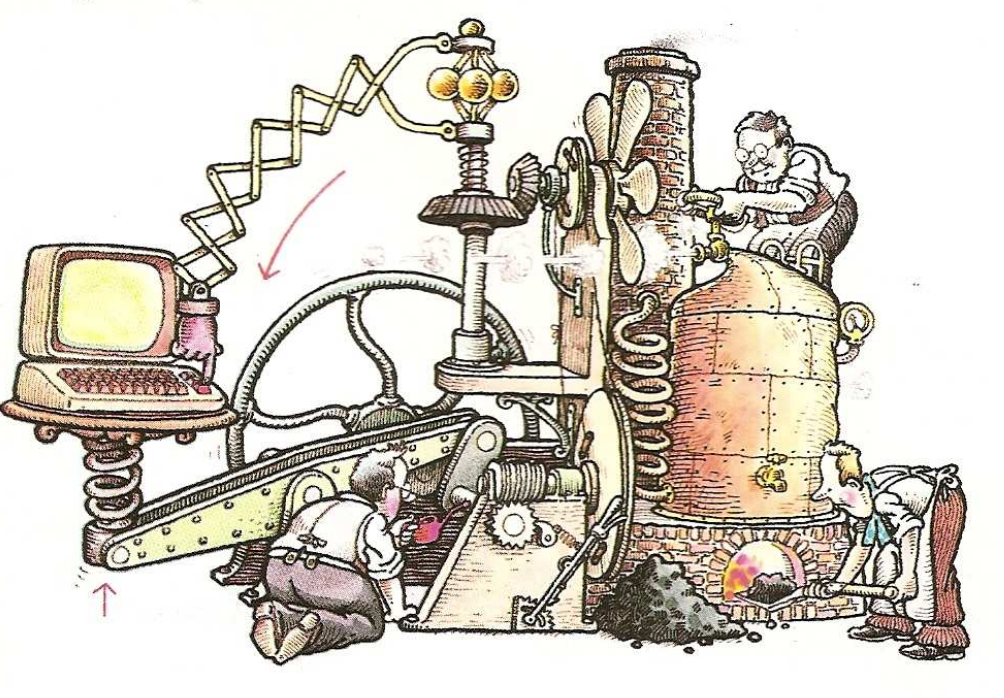
Actualmente serían muy pocos los constructores que se atrevieran a abordar este tipo de trabajos sin disponer de la maquinaria adecuada para cortar y transportar las enormes piedras o para elevar las grandes masas de material a grandes alturas sobre el suelo. Y sin embargo hasta la más sofisticada de dichas máquinas está basada en unos principios fundamentales que ya eran conocidos y utilizados por los ingenieros primitivos. Entre todos ellos destaca especialmente el principio de la mecánica, que es la ciencia que se ocupa del estudio de las fuerzas.

Ya se trate de un sistema estacionario (como un edificio) o en movimiento (como los elementos de una máquina), siempre hay fuerzas que actúan sobre él. Se pueden valorar estas fuerzas de un modo instintivo o a partir de la experiencia, igual que hacían los constructores primitivos, y lo que hacemos todos en las actividades de la vida diaria. Por ejemplo cuando vas a levantar con una mano una taza de café, automáticamente aplicas la fuerza correcta para levantarla sin que el café salga volando. Y si vas a colocar una estantería, tienes una idea bastante precisa del tipo de maderas que tienes que utilizar y de cómo colocar los soportes para evitar que se comben por acción del peso sobre la misma.

La estimación de fuerzas de esta manera está muy bien para las aplicaciones corrientes, pero hay muchas ocasiones en que se requiere una mayor precisión. En este tipo de análisis es donde intervienen los ordenadores.

Naturalmente, puedes utilizar estimaciones basadas en la experiencia para programar simulaciones en ordenador de sistemas en los que intervienen fuerzas. En su nivel más sencillo, esto es lo que se hace en el tipo de simulación empleado en muchos programas de juegos. Así puedes estimar por ejemplo lo lejos que caerá una espada cuando es arrancada de la mano de un enemigo, o con qué fuerza se escuchará el ruido producido por el choque de dos objetos.

Pero cuando utilices como herramienta el ordenador, es seguro que necesitarás mayor precisión. Por ejemplo, tanto si eres un ingeniero que está proyectando un puente como un mecánico aficionado intentando construir un artilugio para levantar el motor de tu coche, es importante saber cuándo se alcanzará el punto de ruptura de la estructura. Si utilizas tu ordenador para analizar las fuerzas que actúan sobre una estructura en condiciones de carga variable, necesitas una precisión absoluta, así como un buen entendimiento de cómo calcular las fuerzas.



LA MAS SENCILLA DE LAS MAQUINAS

A medida que se construyen estructuras más pesadas, el problema de moverlas se va convirtiendo cada vez en un reto mayor. Durante la década de los sesenta, el poderoso cohete **Saturno V** que sirvió para enviar hombres a la Luna, fué transportado sobre la mayor plataforma rodante del mundo. En la actualidad, las plataformas petrolíferas del Mar del Norte, apoyadas sobre colchones neumáticos tipo *hovercraft* o plásticos lubricantes, son remolcadas hasta la playa y botadas al mar. Estas técnicas y otras parecidas dependen para su realización de algu-

na clase de máquinas, que les proporcionan, por ejemplo, tracción, presión o elevación. A su vez estas máquinas son una superposición de unos cuantos dispositivos básicos que han sido utilizados durante siglos.

La más primitiva de estas máquinas es la palanca. En su forma más sencilla, la palanca es una barra rígida, uno de cuyos extremos se coloca debajo de un objeto mientras que se actúa sobre el otro extremo para mover el objeto. Con este simple artificio, una persona puede mover una carga varias veces superior a su propio peso. De hecho, se atribuye a Arquímedes, el gran científico de la Antigüedad, la frase «*Dadme un punto de apoyo y moveré el mundo*» en una clara referencia a la ley de la palanca. Según afirmaba, sería capaz de mover la Tierra siempre que tuviera una palanca suficientemente larga y un buen fulcro. El fulcro o punto de apoyo es el punto alrededor del cual pivota la palanca, por lo que en el ejemplo anterior es simplemente un lugar sobre el suelo en el que se apoya la barra rígida.

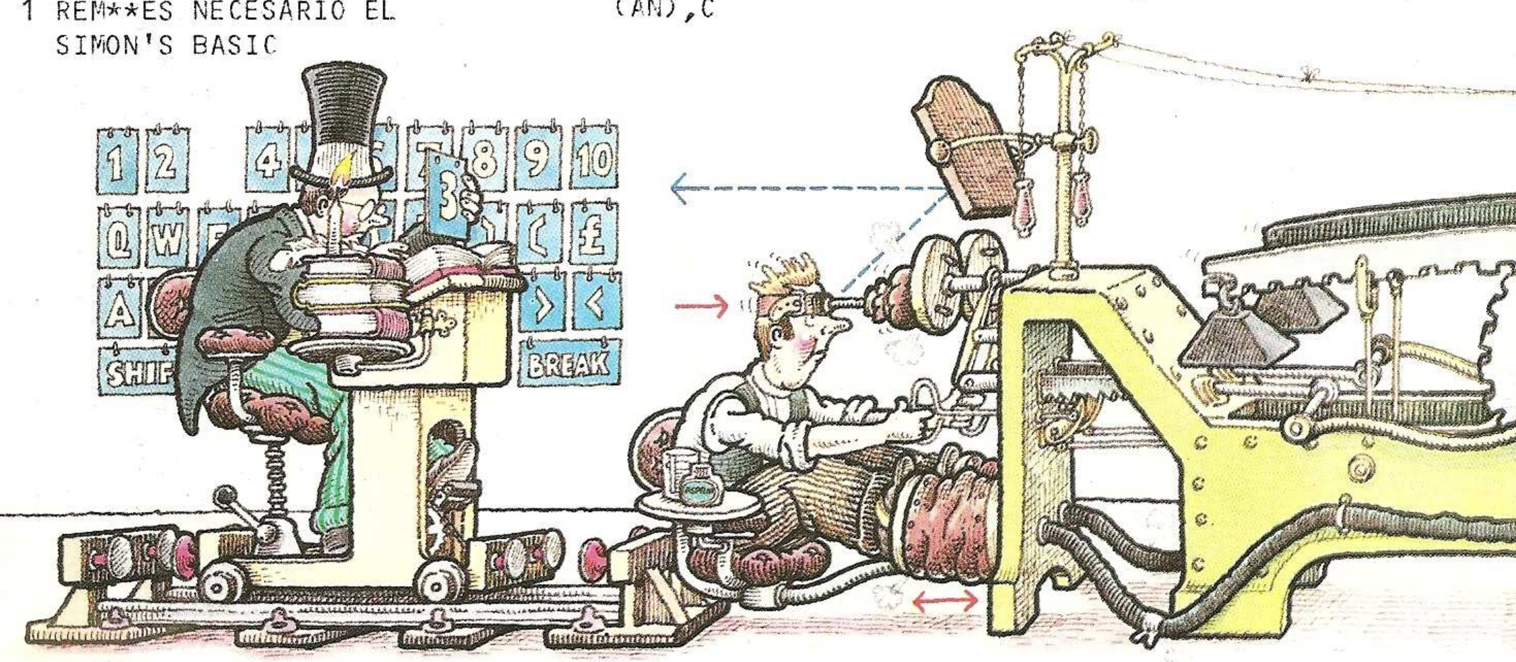
La disposición más normal de la palanca tiene el fulcro no en uno de los extremos de la barra rígida, sino en algún punto entre ambos extremos. Teclea el primer programa que figura a continuación y tendrás una demostración de esto.

Teclea para Commodore-64

```
1 REM**ES NECESARIO EL
  SIMON'S BASIC
10 HIRES 0,3
20 COLOUR 6,3:FOR Z=0 TO 4
   STEP .5:LINE 0,180+Z*Z,
   319,180+Z*Z,1:NEXT Z
40 CSET(0):INPUT "[SHIFT+CLR
   /HOME]DISTANCIA DEL
   FULCRO DESDE LA IZQUIERDA
   (1-8)";D
50 IF D<1 OR D>8 THEN 40
60 W=(10-D)/D
70 PRINT "[CRSR abajo]PESO
   REQUERIDO PARA EQUILIBRAR
   100 KG=",W*100;"KG"
80 FOR G=1 TO 2000:NEXT G:
   CSET(2)
100 LINE 61+20*D,150,51+20*D
    ,179,1:LINE 61+20*D,150,
    71+20*D,179,1
105 PAINT 61+20*D,160,1
120 A=10-D:A=ATN(A*A-1)
130 FOR AN=A TO 2*ATN(1)STEP
    (2*ATN(1)-A)/10
140 C=1:GOSUB 1000:C=1:GOSUB
    1500
150 FOR G=1 TO 500:NEXT G
160 C=0:GOSUB 1000:GOSUB
    1500
170 NEXT AN
180 AN=2*ATN(1):C=1:GOSUB
    1000:GOSUB 1500
190 GET A$:IF A$="" THEN 190
200 RUN
1000 X1=160-100*SIN(AN):Y1=
     149-20*D*COS(AN)
1010 LINE X1,Y1,167+100*SIN
     (AN),149+(200-20*D)*COS
     (AN),C
1020 X1=160+100*SIN(AN):Y1=
     141+(200-20*D)*COS(AN)
1025 BLOCK X1,Y1,166+100*SIN
     (AN),149+(200-20*D)*COS
     (AN),C
1030 RETURN
1500 X1=165-100*SIN(AN):Y1=
     148-20*D*COS(AN)
1510 BLOCK 165-100*SIN(AN)-
     SQR(W)*7,148-20*D*COS
     (AN)-SQR(W)*7,X1,Y1,C
1530 RETURN
```

Al ejecutar el programa aparecerá un mensaje en la pantalla (línea 40) para que especifiques la distancia (D) desde un punto de apoyo hasta el extremo izquierdo de una barra de 10 metros. La línea 60 calcula entonces el peso (W) requerido en dicho extremo para equilibrar una carga de 100 Kg colocada en el otro extremo; la línea 70 presenta en la pantalla el valor de W. Las líneas 100 a 110 se ocupan de dibujar el fulcro. La línea 120 define las variables correspondientes al ángulo de la palanca y las líneas 130 a 170 animan el equilibrio de la barra mediante la llamada a una rutina (líneas 1000 a 1030) para dibujar la barra y otra (líneas 1500 a 1530) para dibujar los pesos. La línea 180 dibuja la posición final de la barra y los pesos.

Prueba a introducir diferentes valores en D cada vez que ejecutes el programa, y observa cómo se requiere un



mayor esfuerzo (es decir un valor mayor de W) a medida que el fulcro va estando más alejado de la carga de 100 Kg. Por el contrario, a medida que el fulcro se va colocando más cerca de la carga, la palanca tiene un efecto mucho mayor, por lo que hace falta mucho menor esfuerzo.

VENTAJA MECANICA

Existe una fórmula matemática muy sencilla para calcular las longitudes y los pesos de un montaje de este tipo, la cual da muy buenos resultados al considerar el efecto de carga sobre cualquier tipo de barra o viga, en el supuesto de que conozcas la posición del punto sobre el que actúa la carga. Dicha fórmula establece que el esfuerzo o potencia multiplicado por la distancia desde el punto donde se aplica hasta el fulcro, es igual a la carga o resistencia multiplicada por su distancia al fulcro. Escribiendo esto con variables podemos poner:

$$E \times DE = C \times DC$$

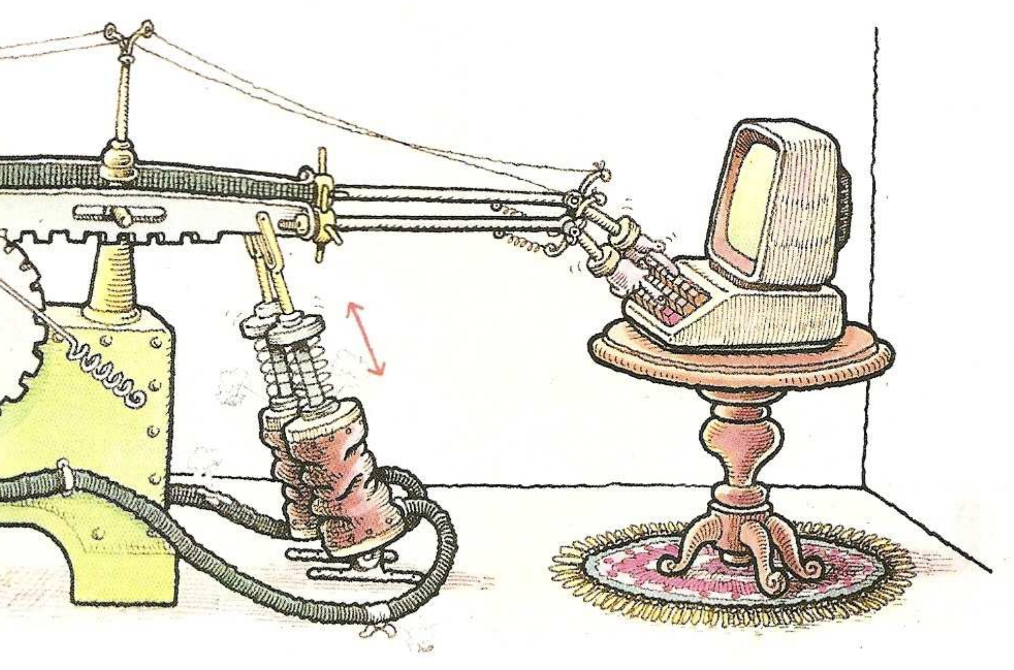
En el programa anterior se utilizó W en lugar de E; la longitud total de la barra es 10 , por lo que DC es igual a 10 − DE. Si el valor de C es igual a 1, la fórmula se convierte en

$$W = (10 - DE)/DE$$

que es la forma en que se utiliza en la línea 60 del programa anterior. Se te pide que especifiques DE (D en el programa), para poder calcular el esfuerzo o potencia a aplicar. Sea cual sea la forma en que utilices la fórmula, siempre podrás calcular la variable que falte si conoces las demás.

Este hecho tan simple se utiliza con gran provecho en las balanzas ordinarias. Como sabes, se trata de una viga de metal que apoya y pivota alrededor de un punto de su centro. Esto significa que DE = DC, ecuación que se puede simplificar para dar E = C. Ahora, para pesar un objeto cualquiera, no tienes más que colocarlo en uno de los platillos que cuelgan de los brazos y poner pesos conocidos en el otro platillo hasta que la balanza quede equilibrada. Esto está muy bien cuando se trata de pesar pequeñas cantidades, como puede ocurrir en una frutería, pero ¿qué sucede cuando hay que pesar un vehículo comercial tal como un camión cargado de mineral o de carbón? De hecho se aplica el mismo principio, pero en este caso en vez de situar el punto de apoyo en el centro se coloca muy cerca de la carga, por lo que se puede seguir equilibrando la balanza con pesos pequeños.

Cuando ya hayas jugado bastante con este programa, y hayas comprobado de qué forma el alargamiento de la palanca disminuye el esfuerzo necesario, estarás en condiciones de descubrir palancas en casi cualquier máquina. Esta disminución del esfuerzo se suele llamar ventaja mecánica y está dada por el cociente entre el esfuerzo y la carga. Si transformas adecuadamente la fórmula anterior, encontrarás que la ventaja mecánica es igual a la distancia que se desplaza el esfuerzo dividida por la distancia que se desplaza la carga. De modo que la próxima vez que tires de una palanca —que puede ser el picaporte de una puerta o un freno de bicicleta— observa cómo el desplazamiento del extremo donde se aplica el esfuerzo es bastante mayor que en el otro extremo.

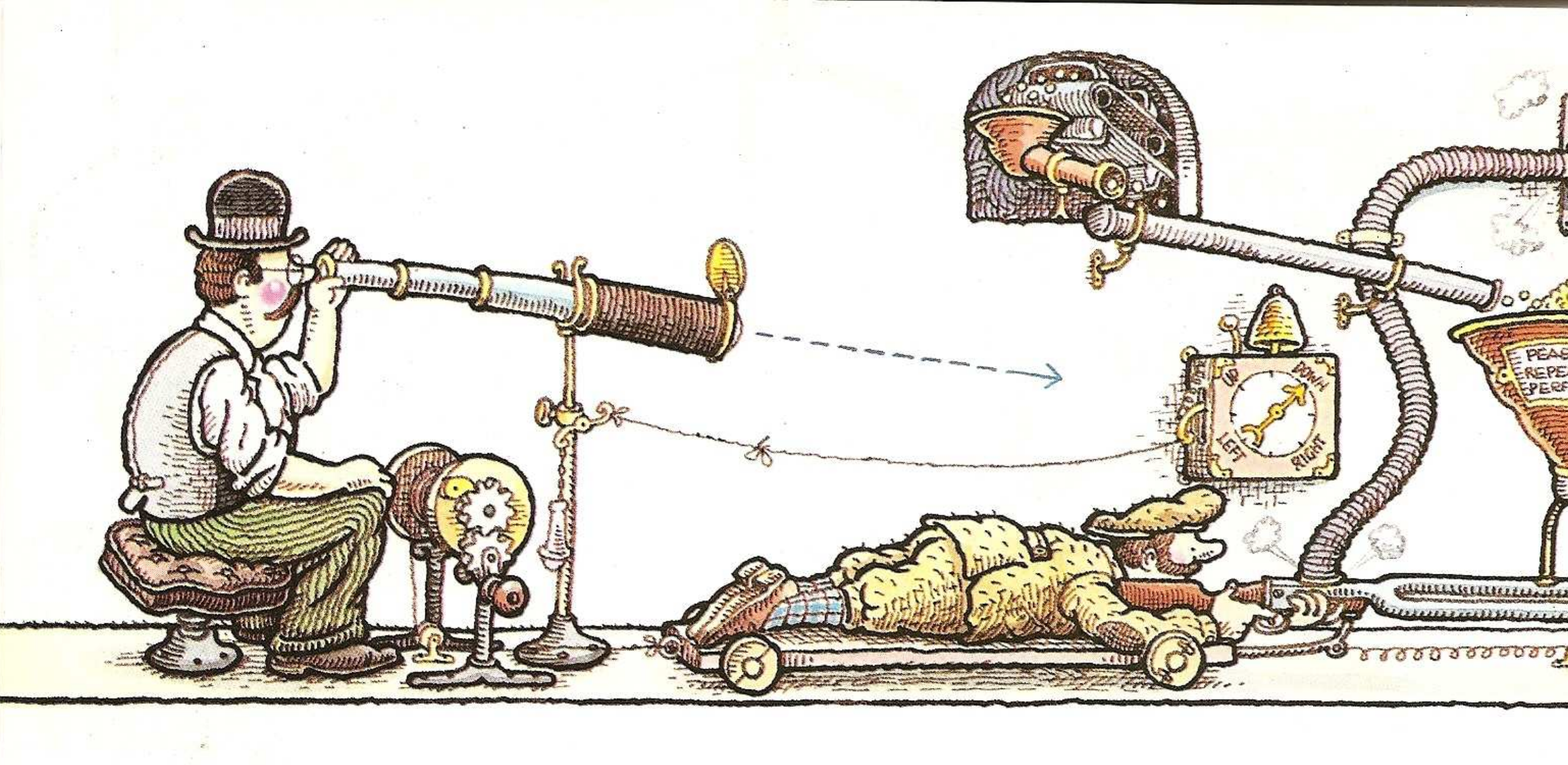


ALCANZANDO MAYORES ALTURAS

Para empujar y tirar de cosas, lo mejor es utilizar palancas, pero cuando lo que hace falta es elevar una determinada carga a una gran altura, la palanca ha de ser modificada. Este es el objeto de los sistemas de poleas, que también te permiten obtener ventaja mecánica. Para ver demostrado esto, teclea y ejecuta el siguiente programa:

Teclea para Commodore-64

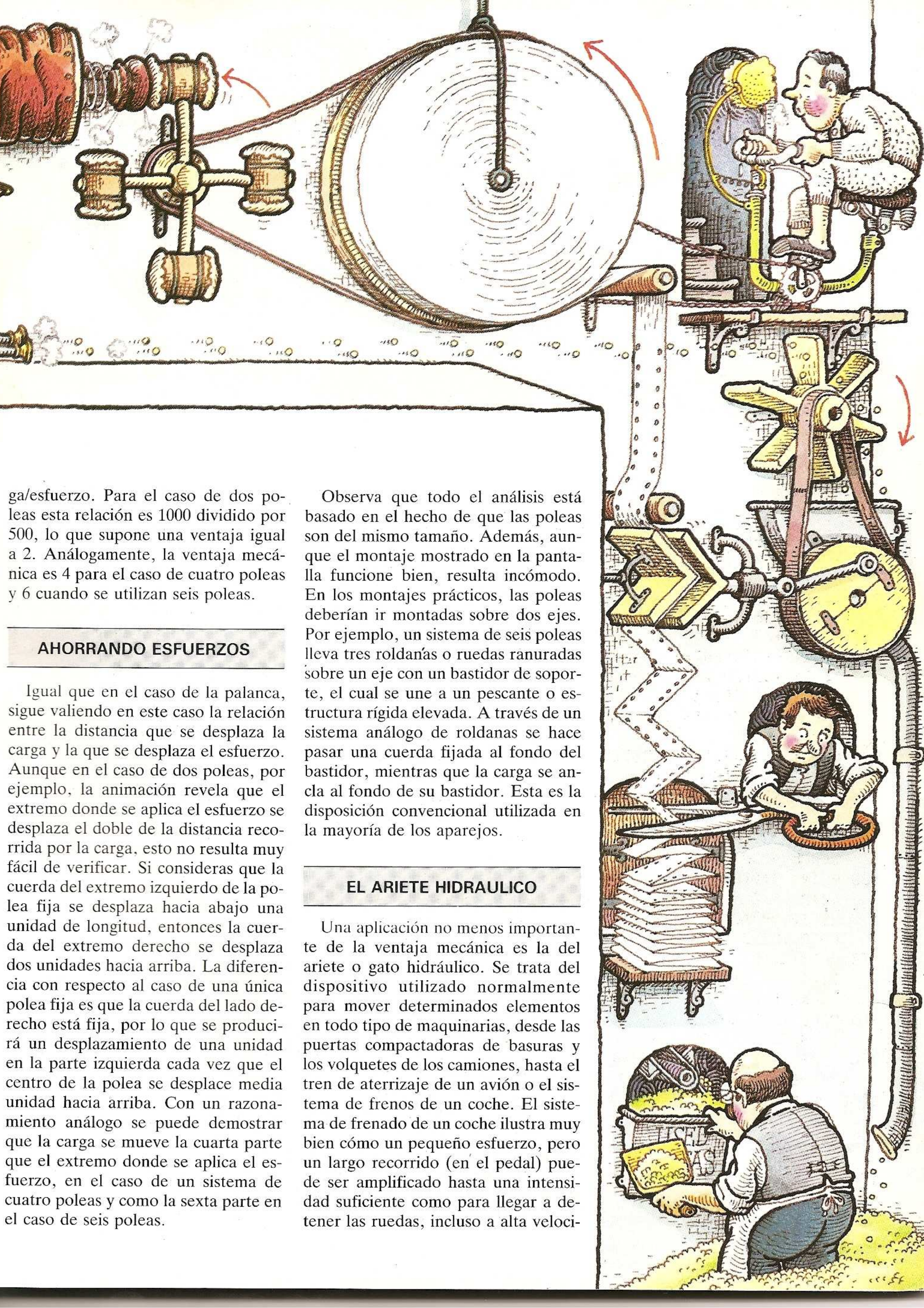
```
1 REM**[]SIGNIFICA ESPACIO
20 A$="[SHIFT+N][CTRL+9][]
   [CTRL+0][SHIFT+M][CRSR
   abajo][3*CRSR izq.]
   [CTRL+9][CRSR dcha.]
   *[CRSR dcha.][CTRL+0]
   [CRSR abajo][3*CRSR izq.]
   [SHIFT+M][CTRL+9][]
   [CTRL+0][SHIFT+N]"
30 FOR Z=1 TO 20:D$=D$+
   "[CRSR abajo]":NEXT Z
40 B$=A$+"[2*CRSR arriba]
   [5*CRSR dcha.]"+A$+"[2*
   CRSR arriba][5*CRSR
   dcha.]"+A$
50 C$="[CTRL+5][CTRL+9][][]
   [CTRL+0][CRSR dcha.]
   [CTRL+9][][][][][][][]
   [CTRL+0][CRSR dcha.][][]
   [][][][][][CTRL+0]
   [CRSR dcha.][CTRL+9][][]
   [CTRL+0]"
60 POKE 53280,7:POKE 53281,7
```




```
    :GOSUB 1000
90  S=0:FOR Z=18 TO 6 STEP-.5
    :PRINT"[CLR/HOME][CTRL+7]
    "LEFT$(D$,Z)TAB(19)B$:
    PRINT"[2*CRSR arriba]"SPC
    (18)C$
100 PRINT TAB(18)"[CTRL+3]
    [SHIFT+J][3*CRSR dcha.]
    [SHIFT+K][][SHIFT+-][]
    [SHIFT+J][3*CRSR dcha.]
    [SHIFT+K][][SHIFT+-][]
    [SHIFT+J][3*CRSR dcha.]
    [SHIFT+K]"
110 PRINT TAB(18)"[][][][][]
    [][COMM.+Z][SHIFT+C]
    [CTRL+1][CTRL+9][]
    [SHIFT+U][][SHIFT+I][]
    [CTRL+0][CTRL+3][SHIFT+C]
    [COMM.+X]"
120 PRINT TAB(24)"[][]
    [CTRL+1][CTRL+9][]
    [SHIFT++][][SHIFT++][]
    [CTRL+0][][]"
130 PRINT TAB(24)"[][][][][]
    [][][][]":FOR D=1 TO 200
    :NEXT D
140 POKE 1278+40*S,93:S=S+3:
    IF S>15 THEN S=0
150 POKE 1278+40*S,220:NEXT
    Z:END
1000 PRINT"[SHIFT+CLR/HOME]
     [CTRL+1]";:FOR Z=1 TO
     40:PRINT "[CTRL+9][]
     [CTRL+0]";:NEXT
1003 FOR Z=1 TO 3:PRINT SPC
     (20)"[CTRL+5][SHIFT+-]
     [7*CRSR dcha.][SHIFT+-]
     [9*CRSR dcha.][SHIFT+-]
     ":NEXT Z
1005 PRINT"[CLR/HOME][3*CRSR
     abajo]"SPC(14)"[SHIFT+U]
     [3*CRSR dcha.][SHIFT+I]
     [3*CRSR dcha.][SHIFT+U]
     [3*CRSR dcha.][SHIFT+I]
     [3*CRSR dcha.][SHIFT+U]
     [3*CRSR dcha.][SHIFT+I]
     [3*CRSR dcha.][SHIFT+-]"
1006 FOR Z=1 TO 16:PRINT
     SPC(14)"[SHIFT+-]
     [3*CRSR dcha.][SHIFT+-]
     [3*CRSR dcha.][SHIFT+-]
     [3*CRSR dcha.][SHIFT+-]
     [3*CRSR dcha.][SHIFT+-]
     [3*CRSR dcha.][SHIFT+-]
     [3*CRSR dcha.][SHIFT+-]"
     :NEXT Z
1007 PRINT SPC(14)"[SHIFT+-]
     [CRSR abajo][CRSR izq.]
     [SHIFT+-][CRSR abajo]
     [CRSR izq.][SHIFT+-]"
1008 PRINT"[2*CRSR arriba]"
     TAB(10)"[CRSR abajo]
     [CTRL+9][CTRL+1][]
     [CTRL+0][COMM+G]"
1010 PRINT"[CLR/HOME][CTRL+7]
     [3*CRSR abajo]"TAB(15)A$
     "[CLR/HOME][3*CRSR
     abajo]"TAB(23)A$"
     [CLR/HOME][3*CRSR abajo]
     "TAB(31)A$
1011 PRINT"[CLR/HOME]
     [4*CRSR abajo]"SPC(14)C$
1020 PRINT"[CLR/HOME][CTRL+6]
     [CRSR abajo]ESFUERZO
     REQUERIDO=":PRINT"166,6
     KILOGRAMOS PARA":PRINT
     "LEVANTAR 1 TONELADA"
1030 RETURN
```

El programa te envía un mensaje para que especifiques el número de poleas que quieres que intervenga en la demostración (línea 20). Para empezar, teclea un 2 y aparecerá en la pantalla (línea 50) el esfuerzo necesario para elevar una carga de 1000 Kg (una tonelada). Seguidamente el programa realiza una animación de la carga cuando está siendo elevada. La rutina que se extiende entre las líneas 1000 y 1210 (a la cual se llama desde la línea 60) sirve para dibujar el sistema de cuerdas, polea fija y apoyos, mientras que las líneas 90 a 150 se ocupan de la polea móvil, la carga y la cuerda que se mueve.

Ejecuta nuevamente el programa pero introduciendo ahora 4 y después 6 como respuesta al mensaje inicial y compara los valores del esfuerzo y de la distancia resultantes para el extremo libre de la cuerda. Como el esfuerzo es igual a la carga dividida por el número de poleas, el esfuerzo resultante será igual a 1/2, 1/4 o 1/6 de los 1000 Kg para los casos de 2, 4 o 6 poleas. En consecuencia, cuanto mayor sea el número de poleas empleadas, menor será el esfuerzo requerido para levantar la misma carga. Al igual que en el caso de la palanca, la ventaja mecánica está dada por la relación car-



ga/esfuerzo. Para el caso de dos poleas esta relación es 1000 dividido por 500, lo que supone una ventaja igual a 2. Análogamente, la ventaja mecánica es 4 para el caso de cuatro poleas y 6 cuando se utilizan seis poleas.

AHORRANDO ESFUERZOS

Igual que en el caso de la palanca, sigue valiendo en este caso la relación entre la distancia que se desplaza la carga y la que se desplaza el esfuerzo. Aunque en el caso de dos poleas, por ejemplo, la animación revela que el extremo donde se aplica el esfuerzo se desplaza el doble de la distancia recorrida por la carga, esto no resulta muy fácil de verificar. Si consideras que la cuerda del extremo izquierdo de la polea fija se desplaza hacia abajo una unidad de longitud, entonces la cuerda del extremo derecho se desplaza dos unidades hacia arriba. La diferencia con respecto al caso de una única polea fija es que la cuerda del lado derecho está fija, por lo que se producirá un desplazamiento de una unidad en la parte izquierda cada vez que el centro de la polea se desplace media unidad hacia arriba. Con un razonamiento análogo se puede demostrar que la carga se mueve la cuarta parte que el extremo donde se aplica el esfuerzo, en el caso de un sistema de cuatro poleas y como la sexta parte en el caso de seis poleas.

Observa que todo el análisis está basado en el hecho de que las poleas son del mismo tamaño. Además, aunque el montaje mostrado en la pantalla funcione bien, resulta incómodo. En los montajes prácticos, las poleas deberían ir montadas sobre dos ejes. Por ejemplo, un sistema de seis poleas lleva tres roldanas o ruedas ranuradas sobre un eje con un bastidor de soporte, el cual se une a un pescante o estructura rígida elevada. A través de un sistema análogo de roldanas se hace pasar una cuerda fijada al fondo del bastidor, mientras que la carga se ancla al fondo de su bastidor. Esta es la disposición convencional utilizada en la mayoría de los aparejos.

EL ARIETE HIDRAULICO

Una aplicación no menos importante de la ventaja mecánica es la del ariete o gato hidráulico. Se trata del dispositivo utilizado normalmente para mover determinados elementos en todo tipo de maquinarias, desde las puertas compactadoras de basuras y los volquetes de los camiones, hasta el tren de aterrizaje de un avión o el sistema de frenos de un coche. El sistema de frenado de un coche ilustra muy bien cómo un pequeño esfuerzo, pero un largo recorrido (en el pedal) puede ser amplificado hasta una intensidad suficiente como para llegar a detener las ruedas, incluso a alta veloci-

dad. Teclea y ejecuta el siguiente programa para ver la demostración de este principio:

```
1 REM**ES NECESARIO EL
  SIMON'S BASIC
10 HIRES 0,1
20 MULTI 0,5,6
30 COLOUR 2,7
40 DIM P(4),R(31)
50 GOSUB 300
80 GET A$:IF A$=""THEN 80
90 CSET (0):INPUT "[SHIFT+
   CLR/HOME]CUAL ES EL
   DESPLAZAMIENTO DEL
   EMBOLO (1-90)";TR
100 IF TR<1 OR TR>90 THEN 90
110 CSETr(2):MULTIr 0,5,6:
    FOR K=1 TO TR
115 LINE 16,40+K,20,40+K,0
120 BLOCK 16,41+K,20,48+K,2
125 LINE 91,53-K/7,111,53-K
    /7,3
130 BLOCK 91,23-K/7,111,48-K
    /7,2
140 NEXT K
150 GET A$:IF A$="" THEN 150
160 CSET(0):PRINT "[SHIFT+
    CLR/HOME]OTRA VEZ (S/N)?"
170 GET A$:IF A$<>"S" AND A$
    <>"N" THEN 170
180 IF A$="S" THEN RUN
190 PRINT "[SHIFT+CLR/HOME]"
    :END
300 LINE 15,40,15,200,1
305 LINE 112,40,112,200,1
310 BLOCK 21,40,90,180,1
330 TEXT 50,170,">",0,1,8
340 BLOCK 16,41,20,48,2
350 BLOCK 91,23,111,48,2
355 PAINT 80,190,3:LINE 55,
    180,55,200,0
360 FOR K=0 TO 9
370 LINE 13,49+K*10,15,49+K
    *10,1
380 NEXT K
390 TEXT 5,46,"0",1,1,1
395 TEXT 0,140,"90",1,1,8
400 FOR K=0 TO 9 STEP 3
410 LINE 113,49-K,115,49-K,1
420 NEXT K
430 TEXT 116,48,"0",1,1,8
435 TEXT 116,35,"9",1,1,8
440 RETURN
```

Al ejecutar este programa, en la línea 50 se produce un salto a una subrutina que dibuja el ariete. A continuación el programa te pide que especifiques el recorrido del pistón, que en un sistema de frenado sería equivalente al movimiento del pedal. Se produce entonces una animación que describe el movimiento del líquido en el ariete.

El desplazamiento es una medida del esfuerzo aplicado al pistón, pero la ventaja mecánica está determinada por los diámetros de los pistones motriz y de carga. Si los dos pistones tienen el mismo diámetro, no hay ventaja mecánica, pero a medida que se reduce el diámetro del pistón motriz aumenta ésta, de forma que es más fácil elevar la carga del lado derecho.

LA CAJA DE HERRAMIENTAS

- INCORPORACION DE NUEVOS COMANDOS
- RUTINA DE RENUMERACION
- NUMERACION AUTOMATICA DE LINEAS
- SUPRESION DE BLOQUES
- OTROS COMANDOS

Haz que tu vida resulte un poco más fácil con este nuevo juego de herramientas de programación. Esta utilidad en código máquina ofrece muchas rutinas que te ayudarán a configurar tus programas.

Aunque todos los ordenadores utilizan en principio el mismo lenguaje BASIC, te habrás dado cuenta de las variantes que existen de unos dialectos a otros. De hecho, es muy raro encontrar un programa, por corto que sea, que corra en más de un ordenador. A veces no hace falta modificar más que la forma en que se utilizan los comandos, pero muchas veces te encontrarás que muchos comandos implementados en uno de los ordenadores no existen en absoluto en el otro. Muchos de los comandos no disponibles son precisamente los que, sin ser absolutamente esenciales para la programación, hacen que la vida resulte más fácil para el programador, por incluir facilidades como la renumeración, la autonumeración de líneas, o la conversión de hexadecimal a decimal, etc.

El programa de caja de herramientas (*toolkit*) para el **Commodore** que vamos a ver a continuación, incorpora estos comandos y algunos otros, facilitándote grandemente el uso y la programación del ordenador. Hay otros ordenadores que no necesitan estas utilidades, porque disponen de ellas en su BASIC estándar.

El juego de utilidades que aquí presentamos incorpora un total de 43 nuevos comandos que pueden utilizarse de la misma forma que las palabras reservadas del BASIC.

Teclea el programa y a continuación ejecútalo para comprobar si hay errores. Si te has equivocado en alguna de las líneas de DATA, el programa te dirá en qué línea están los errores.

Cuando el programa es correcto, almacena la versión de BASIC por medio de:

```
SAVE"HERRAMIENTA"
```

A continuación teclea RUN y después SYS 52480, para crear el código máquina; después almacena dicho código máquina con uno de los nuevos comandos recién creados.

```
@MSAVE 49152,53247,"HERRA
MIENTA CM",1,1
```

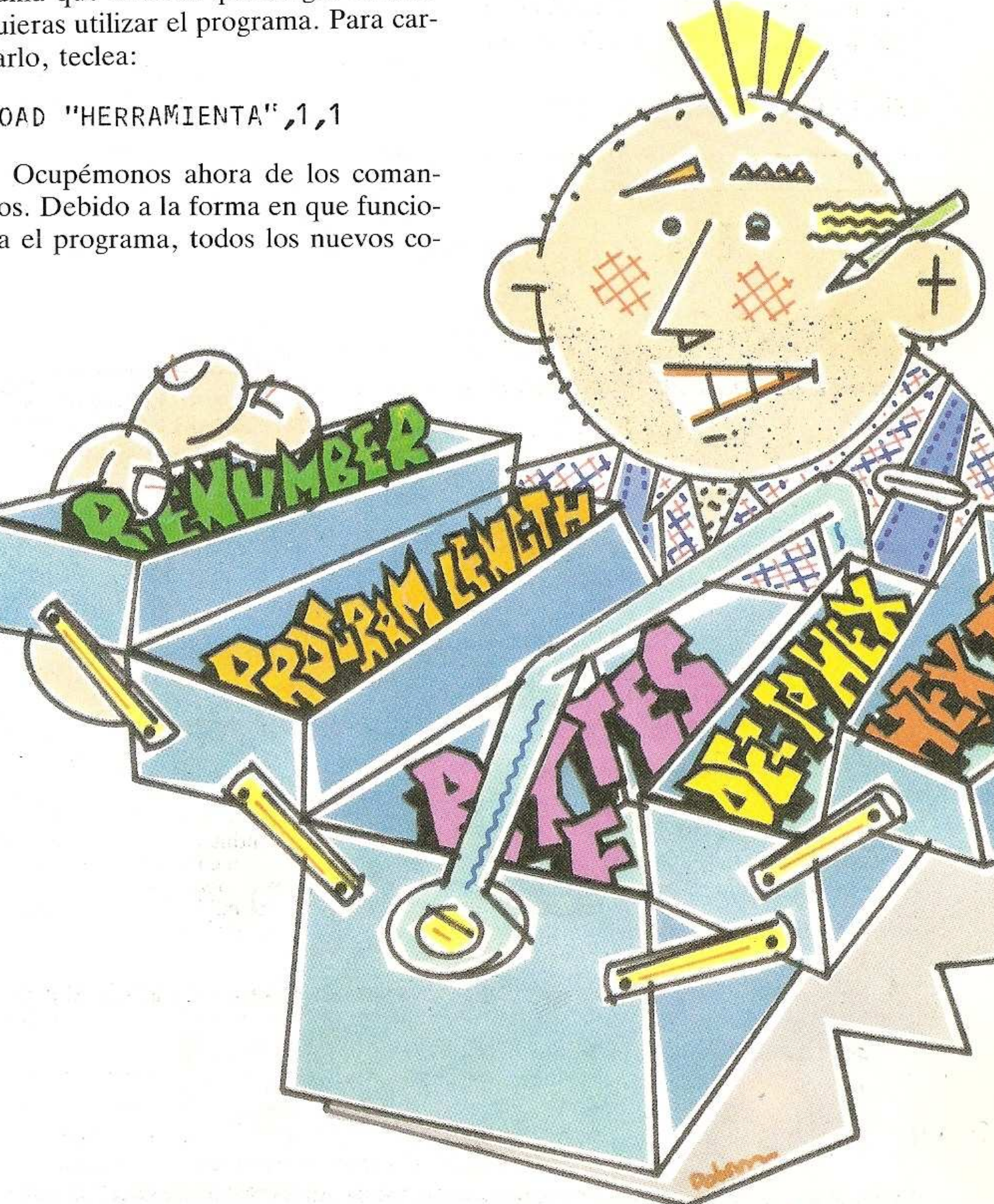
Se trata de la versión del código máquina que tendrás que cargar cuando quieras utilizar el programa. Para cargarlo, teclea:

```
LOAD "HERRAMIENTA",1,1
```

Ocupémonos ahora de los comandos. Debido a la forma en que funciona el programa, todos los nuevos comandos se componen de una palabra de BASIC ya existente más un par de caracteres adicionales, el primero de los cuales siempre será @. La sintaxis de cada comando va seguida de una explicación así como de un ejemplo en los casos en que sea necesario.

LOS NUEVOS COMANDOS

@PPOKE posición, dirección. Por ejemplo @PPOKE 51, 16384 POKEa



@POKER dirección de comienzo, número de elementos (0 a 256), lista de elementos. Por ejemplo @POKER 49152,4,12,51,34,15,21 POKEa los cinco datos en la memoria a partir de la dirección 49152.

@POKES dirección inicial, dirección final, valor a POKEar (0 a 255).



Este comando llena toda una sección de memoria con el valor deseado. Puedes borrar una sección de memoria POKEando ceros, o también puedes llenar parte de la pantalla con caracteres, POKEando en la memoria de pantalla el código del carácter que quieres que aparezca en pantalla.

los bytes bajo y alto de 16384 en las posiciones de memoria 51 y 52.

@PPEEK posición. Por ejemplo @PPEEK 51 presenta el valor de PEEK(51) + PEEK(52)*256.

@BNEW dirección. Desplaza el comienzo del BASIC a la dirección deseada.

@GETNEW resetea el comienzo del BASIC a su valor normal. Es equivalente a un comando OLD. Puedes utilizar los dos últimos comandos para almacenar varios programas en memoria al mismo tiempo. Por ejemplo, puedes cargar un programa como de costumbre y a continuación desplazar el BASIC, por ejemplo con @BNEW 16384, cargando después otro programa. @BNEW 2048:@GETNEW te hará volver al programa antiguo.

@NNEW realiza un arranque frío, borrando todos los programas de la memoria.

@GGOTO variable te permite hacer un GOTO a un número de línea variable.

@LRESTORE número de línea te permite restaurar un comando READ a una línea específica.

@MGOTO dirección inicial, dirección final, nuevo comienzo desplaza la zona de memoria situada entre las direcciones inicial y final, llevándola hacia la nueva dirección de comienzo.

@MREAD dirección inicial, dirección final imprime el contenido de la zona de memoria sumándole el total de todos los valores. Un resultado típico podría ser 12, 32, 65, 22, #131. Se puede utilizar este comando para crear líneas DATA de código máquina con comprobaciones al final de cada línea. Al final de la sección hay un corto programa para demostrar esto.

@MSAVE dirección inicial, dirección final, nombre de programa en código máquina, número de dispositivo, 1 almacena una zona de memoria como programa en código máquina. Como ejemplo, puedes ver el comando para almacenar el programa del juego de utilidades.

@MFRE te informa del espacio total de memoria disponible para el BASIC.

@COST byte bajo, byte alto. Por ejemplo @COST 6,1 presenta el valor de 6 + 1*256.

@POST número convierte el número en sus bytes bajo y alto.

@CCHR$ número de caracteres (0 a 255), código ASCII a imprimir. Por ejemplo, @CCHR$ 6,65 imprime seis letras A.

@D' número decimal, lo convierte a hexadecimal.

@H'$ número hexadecimal de cuatro dígitos, lo convierte a decimal.

@ONTO incremento de cuenta (0 a 255) proporciona numeración automática de líneas con el incremento deseado. La línea de comienzo es la primera que teclees.

@ONTO RETURN anula el efecto del anterior comando @ONTO. Antes de emplear este comando, pulsa SHIFT y RETURN .

@RLIST, número de línea de comienzo, incremento. Es es comando de renumeración. Por ejemplo, @RLIST,10,5 renumera un programa empezando en la línea 10 y avanzando de 5 en 5. Fíjate en la primera coma que figura en el comando y cuida que no se te olvide.

@WAITGET variable, espera a que se produzca una pulsación de tecla; el nombre de la variable carece de importancia.

@WAIT' 711*número de segundos introduce un tiempo de espera del número de segundos especificado (el 711 es un factor de escala). @PRINT% tinta (0 a 15), X (0 a 39), Y (0 a 24), texto a imprimir. Este comando imprime el texto deseado, en el color especificado y en las coordenadas X, Y.

@COR tinta (0 a 15), color de borde (0 a 15), color de papel (0 a 15) establece unas determinadas condiciones de color.

@CLR' número (0 a 39) borra de la pantalla la línea especificada.

@SCLR borra toda la pantalla.

@NEW resetea la pantalla.

@UP desplaza toda la pantalla una línea hacia arriba.

@SYS1 deja la pantalla inactiva.

@SYS0 pone la pantalla activa.

@ASC1 hace que todas las letras sean minúsculas.

@ASC0 hace que todas las letras sean mayúsculas.

@ON1 inhibe la tecla SHIFT.

@ON0 activa la tecla SHIFT.

@DEF1 inhibe la tecla RUN/STOP.

@DEF0 activa la tecla RUN/STOP.

@FN1 activa la autorrepetición.

@FN0 desactiva la autorrepetición.

@KCLR borra el *buffer* de teclado (es equivalente a POKE 198,0).

@TOP envía el cursor a la esquina superior izquierda de la pantalla.

@SIF borra todos los registros del *chip* de sonido.

@SON voz, volumen, A/D,S/R, forma de onda, byte alto de la nota, byte bajo de la nota. Puedes utilizar este comando para definir los parámetros de una nota.

CREACION DE LINEAS DATA

Tal como hemos mencionado al hablar del comando @MREAD, leyendo una zona de memoria, puedes crear un programa BASIC que consista en sentencias DATA de código máquina. Este es el comando que se utilizó para crear el listado del propio paquete de utilidades. Suponiendo que has utilizado un ensamblador para crear un determinado código máquina que empiece en la posición 49152, el método es el siguiente. En primer lugar, teclea las siguientes líneas y a continuación pulsa RETURN :

```
A=0:X=49152:@CCHR$79,32:?
"[2*CRSR arriba][]"
A*10"DATA[]";:@MREAD X+A
*15,14+X+A*15
```

Al pulsar RETURN te aparecerá en la pantalla la primera línea DATA. Coloca el cursor sobre dicha línea y pulsa RETURN para almacenarla.

A continuación, desplaza el cursor a A = 0 e incrementa el 0 a 1, pulsa RETURN y sigue el proceso con la siguiente línea de datos. Sigue incrementando A hasta que estén introducidos todos los datos. Esta rutina te presenta un programa con incrementos de línea de 10 en 10. Si cambias el número 10 que precede a «DATA», obtendrás otros incrementos distintos.

SUPRESION DE BLOQUES

No existe un comando para suprimir bloques, pero se puede obtener

este mismo efecto por medio de otros dos comandos. Activa la función de autorrepetición por medio de @FN1 y a continuación teclea @ONTO con un valor adecuado de incremento de línea, relacionado con el incremento entre líneas que deseas suprimir; finalmente, teclea el número de la primera línea que ha de ser suprimida. Al mantener pulsado RETURN se borrarán efectivamente todas las líneas desde ese punto en adelante, deteniéndose únicamente cuando dejes de apretar la tecla RETURN .

```
100 DATA 32,158,183,142,134,
    2,32,253,174,32,158,183,
    138,72,32,#1725
101 DATA 253,174,32,158,183,
    104,168,24,32,240,255,32
    ,253,174,32,#2114
102 DATA 164,170,96,32,158,
    183,142,134,2,32,253,174
    ,32,158,183,#1913
103 DATA 142,32,208,32,253,
    174,32,158,183,142,33,
    208,96,32,247,#1972
104 DATA 183,32,253,174,32,
    235,183,142,19,3,169,0,
    133,2,32,#1592
105 DATA 253,174,32,158,183,
    138,164,2,145,20,204,19,
    3,240,5,#1740
106 DATA 230,2,76,74,192,96,
    32,138,173,32,247,183,
    165,20,133,#1793
107 DATA 251,165,21,133,252,
    32,253,174,32,138,173,32
    ,247,183,32,#2118
108 DATA 253,174,32,158,183,
    134,2,165,21,197,252,144
    ,35,208,6,#1964
109 DATA 165,20,197,251,144,
    27,165,2,160,0,145,251,
    165,251,197,#2140
110 DATA 20,208,6,165,252,
    197,21,240,9,230,251,208
    ,234,230,252,#2523
111 DATA 76,141,192,96,32,
    138,173,32,247,183,165,
    20,133,251,165,#2044
112 DATA 21,133,252,32,253,
    174,32,138,173,32,247,
    183,165,20,133,#1988
113 DATA 253,165,21,133,254,
    32,253,174,32,138,173,32
    ,247,183,165,#2255
114 DATA 254,197,252,144,41,
    208,6,165,253,197,251,
    144,33,160,0,#2305
115 DATA 177,251,145,20,165,
    251,197,253,208,6,165,
    252,197,254,240,#2781
116 DATA 15,230,251,208,2,
    230,252,230,20,208,228,
    230,21,76,223,#2424
117 DATA 192,96,32,138,173,
    32,247,183,165,20,133,
    251,165,21,133,#1981
118 DATA 252,32,253,174,32,
    138,173,32,247,183,160,0
    ,165,20,145,#2006
119 DATA 251,165,21,200,145,
    251,96,32,158,183,32,255
    ,233,96,32,#2150
120 DATA 138,173,32,247,183,
    160,0,177,20,170,200,177
    ,20,32,205,#1934
121 DATA 189,96,32,138,173,
    32,247,183,165,20,133,
    251,165,21,133,#1978
122 DATA 252,32,253,174,32,
    138,173,32,247,183,165,
    21,197,252,144,#2295
123 DATA 61,208,6,165,20,197
    ,251,144,53,169,0,133,
    253,133,254,#2047
124 DATA 24,160,0,177,251,
    170,101,253,133,253,165,
    254,105,0,133,#2179
125 DATA 254,169,0,32,205,
    189,169,44,32,210,255,
    165,251,197,20,#2192
126 DATA 208,6,165,252,197,
    21,240,9,230,251,208,
    214,230,252,76,#2559
127 DATA 104,193,169,35,32,
    210,255,166,253,165,254,
    76,205,189,32,#2338
128 DATA 138,173,32,247,183,
    162,0,232,208,253,198,20
    ,169,255,197,#2467
129 DATA 20,208,245,198,21,
    197,21,208,239,96,32,158
    ,183,160,0,#1986
```

```
130 DATA 224,1,208,2,160,7,
    224,2,208,2,160,14,132,
    2,32,#1378
131 DATA 253,174,32,158,183
    ,142,24,212,32,253,174,
    32,158,183,138,#2148
132 DATA 164,2,153,5,212,32,
    253,174,32,158,183,138,
    164,2,153,#1825
133 DATA 6,212,32,253,174,32
    ,158,183,138,164,2,153,4
    ,212,142,#1865
134 DATA 19,3,32,253,174,32,
    158,183,138,164,2,153,1,
    212,32,#1556
135 DATA 253,174,32,158,183,
    138,164,2,153,0,212,206,
    19,3,173,#1870
136 DATA 19,3,164,2,153,4,
    212,96,162,0,138,157,0,
    212,232,#1554
137 DATA 224,25,208,248,96,
    76,68,229,76,24,229,76,
    234,232,169,#2214
138 DATA 0,141,138,2,96,169,
    128,141,138,2,96,169,0,
    133,198,#1551
139 DATA 96,169,237,141,40,3
    ,96,169,251,141,40,3,96,
    76,102,#1660
140 DATA 229,169,27,141,17,
    208,96,169,11,141,17,208
    ,96,169,21,#1719
141 DATA 141,24,208,96,169,
    23,141,24,208,96,169,9,
    76,210,255,#1849
142 DATA 169,8,76,210,255,32
    ,138,173,32,247,183,76,
    163,168,32,#1962
143 DATA 138,173,32,247,183,
    169,0,168,145,20,24,165,
    20,105,1,#1590
144 DATA 133,43,165,21,105,0
    ,133,44,76,154,227,169,
    62,32,210,#1574
145 DATA 255,169,18,32,210,
    255,165,55,56,229,45,170
    ,165,56,229,#2109
146 DATA 46,32,205,189,169,
    96,160,228,76,30,171,169
    ,0,133,198,#1902
147 DATA 165,198,201,1,208,
    250,76,146,171,169,8,160
    ,1,145,43,#1942
148 DATA 32,51,165,24,165,34
    ,105,2,133,45,133,47,133
    ,49,165,#1283
149 DATA 35,105,0,133,46,133
    ,48,133,50,96,32,138,173
    ,32,247,#1401
150 DATA 183,165,20,133,63,
    165,21,133,64,32,19,166,
    56,165,95,#1480
151 DATA 233,1,133,65,165,96
    ,233,0,133,66,96,162,0,
    181,43,#1607
152 DATA 149,151,232,224,4,
    208,247,32,138,173,32,
    247,183,165,20,#2205
153 DATA 133,43,165,21,133,
    44,32,253,174,32,138,173
    ,32,247,183,#1803
154 DATA 165,20,133,45,165,
    21,133,46,32,86,225,162,
    0,181,251,#1665
155 DATA 149,43,232,224,4,
    208,247,96,32,158,183,
    134,2,32,253,#1997
156 DATA 174,32,158,183,142,
    19,3,165,2,201,0,240,11,
    173,19,#1522
157 DATA 3,32,210,255,198,2,
    76,79,195,96,76,154,227,
    32,138,#1773
158 DATA 173,32,247,183,170,
    169,72,32,210,255,169,39
    ,32,210,255,#2248
159 DATA 169,36,32,210,255,
    138,32,139,195,138,32,
    144,195,152,32,#1899
160 DATA 139,195,152,32,144,
    195,96,24,106,106,106,
    106,41,15,24,#1481
161 DATA 105,48,201,58,144,
    2,105,6,32,210,255,96,
    169,68,32,#1531
162 DATA 210,255,169,39,32,
    210,255,32,186,195,133,
    34,32,186,195,#2163
163 DATA 170,165,34,32,205,
    189,76,228,167,32,203,
    195,10,10,10,#1726
164 DATA 10,133,35,32,203,
    195,101,35,133,35,96,32,
    115,0,201,#1356
```

GANADORES DE LOS MEJORES DE INPUT COMMODORE

En el sorteo correspondiente al número 12 entre quienes escribisteis mandando vuestros votos a LOS MEJORES DE INPUT han resultado ganadores:

NOMBRE	LOCALIDAD	JUEGO ELEGIDO
Marcos Durán Serrano	Barcelona	World Cup Carnival
Manuel Revuelta Sierra	Alcobendas (Madrid)	Spindizzy
Luis Cazorla Seisdedos	H. de Llobregat (Barcelona)	Games Maker
Jaime Fernández del Mazo	Madrid	Sauccer attack II
Rubén Igual Valero	Barcelona	Green Beret
David Lumbreras Boldova	Zaragoza	Desert Fox
Rafael Pérez Fraile	Santander	RMS - Titanic
Ramón Pedro Ñeco García	Alicante	Skyfox
J. Manuel Carrillo Montero	Córdoba	Hardball
Carlos Morata Sole	Lérida	Rambo

```
165 DATA 58,41,15,144,2,105,
    8,96,32,138,173,32,247,
    183,169,#1443
166 DATA 91,32,210,255,169,0
    ,166,20,32,205,189,169,
    44,32,210,#1824
167 DATA 255,169,0,166,21,32
    ,205,189,169,93,32,210,
    255,96,32,#1924
168 DATA 158,183,134,2,32,
    253,174,32,158,183,138,
    166,2,76,205,#1896
169 DATA 189,5,0,0,32,121,0,
    208,6,169,0,141,14,196,
    96,#1177
170 DATA 169,1,141,14,196,
    169,53,141,4,3,169,196,
    141,5,3,#1405
171 DATA 32,138,173,32,247,
    183,165,20,141,12,196,
    165,21,141,13,#1679
172 DATA 196,96,173,0,2,201,
    48,144,59,201,58,176,55,
    173,14,#1596
173 DATA 196,240,50,32,124,
    165,132,2,173,12,196,24,
    101,20,133,#1600
174 DATA 99,173,13,196,101,
    21,133,98,162,144,56,32,
    73,188,32,#1521
175 DATA 223,189,133,254,132
    ,244,160,0,177,254,240,6
    ,153,119,2,#2286
176 DATA 200,208,246,132,198
    ,164,2,96,76,124,165,0,0
    ,0,0,#1611
177 PRINT"[SHIFT+CLR/HOME]":
    X=49152:C=76:GOSUB 200
178 DATA 169,11,141,8,3,169,
    205,141,9,3,96,32,115,0,
    201,#1303
179 DATA 64,240,3,76,231,167
    ,160,1,177,122,133,255,
    160,2,177,#1968
180 DATA 122,133,2,162,0,189
    ,128,206,197,255,208,9,
    232,189,128,#2160
181 DATA 206,197,2,240,16,
    202,201,0,240,6,232,232,
    224,128,208,#2334
182 DATA 230,162,11,108,0,3,
    134,2,32,115,0,32,115,0,
    32,#976
183 DATA 115,0,166,2,189,255
    ,205,133,252,189,0,206,
    133,253,169,#2267
184 DATA 76,133,251,32,251,0
    ,76,174,167,96,0,0,0,0,0
    ,#1256
185 X=52480:C=6:GOSUB 200
186 DATA 0,192,33,192,64,192
    ,96,192,169,192,1,193,36
    ,193,43,#1788
187 DATA 193,61,193,163,193,
    189,193,36,194,48,194,51
    ,194,54,194,#2150
188 DATA 57,194,63,194,69,
    194,74,194,80,194,86,194
    ,89,194,95,#1971
189 DATA 194,101,194,107,194
    ,113,194,118,194,123,194
    ,132,194,159,194,#2405
190 DATA 189,194,202,194,233
    ,194,8,195,65,195,97,195
    ,100,195,159,#2415
191 DATA 195,215,195,251,195
    ,15,196,73,197,0,0,0,0,0
    ,0,#1532
192 X=52736:C=5:GOSUB 200
193 DATA 153,37,67,176,151,
    82,151,83,77,137,80,151
    ,156,39,80,#1620
194 DATA 194,77,135,146,39,
    83,145,83,139,83,156,83
    ,162,85,80,#1690
195 DATA 165,48,165,49,75,
    156,150,48,150,49,164,
    80,158,48,158,#1663
196 DATA 49,198,48,198,49,
    145,48,145,49,71,137,66,
    162,77,184,#1626
197 DATA 146,161,161,162,76,
    140,77,148,67,199,78,162
    ,69,39,72,#1757
198 DATA 39,185,84,190,84,
    145,164,82,155,0,0,0,0,0
    ,0,#1128
199 X=52864:C=5:GOSUB 200:
    GOTO 205
200 FOR Z= 0 TO C:T=0:FOR ZZ
    =0 TO 14:READ M:POKE X,M
201 PRINT"[CRSR arriba]LINEA
    "PEEK(63)+PEEK(64)*256:
    T=T+M:X=X+1
202 NEXT ZZ:READ X$:IF VAL(
    RIGHT$(X$,LEN(X$)-1))<>T
    THEN 204
203 NEXT Z:PRINT"OK[2*CRSR
    abajo]":RETURN
204 PRINT"ERROR EN LA LINEA"
    :END
205 K=50505:T=0
206 READ A:IF A=-1 THEN 209
207 POKE K,A:K=K+1
208 T=T+A:GOTO 206
209 IF T<>52549 THEN PRINT
    "[2*CRSR abajo][]ERROR
    EN LA SUMA DE COMPROBA
    CION":END
210 IF K<>50928 THEN PRINT
    "[2*CRSR abajo]ERROR EN
    LAS CIFRAS":END
211 PRINT"USA SYS 52480 PARA
    EJECUTAR EL CODIGO
    MAQUINA"
212 END
213 DATA 32,253,174,32,107,
    169,165
214 DATA 20,133,53,165,21,
    133,54
215 DATA 32,253,174,32,107,
    169,165
216 DATA 20,133,49,165,21,
    133,50
217 DATA 32,142,166,32,201,
    198,32
218 DATA 201,198,208,33,32,
    2,198
219 DATA 32,201,198,32,201,
    198,208
220 DATA 3,76,212,198,32,201
    ,198
221 DATA 165,99,145,122,32,
    201,198
222 DATA 165,98,145,122,32,
    13,198
223 DATA 240,226,32,201,198,
    32,201
224 DATA 198,32,201,198,201,
    34,208
225 DATA 11,32,201,198,240,
    197,201
226 DATA 34,208,247,240,238,
    170,240
227 DATA 188,16,233,162,4,
    221,235
228 DATA 198,240,5,202,208,
    248,240
229 DATA 221,165,122,133,59,
    165,123
230 DATA 133,60,32,115,0,
    176,211
231 DATA 32,107,169,32,32,
    198,165
232 DATA 60,133,123,165,59,
```

```
     133,122
233 DATA 160,0,162,0,189,0,1
234 DATA 240,17,72,32,115,0,
     144
235 DATA 3,32,82,198,104,
     160,0
236 DATA 145,122,232,208,234
     ,32,115
237 DATA 0,176,8,32,97,198,
     32
238 DATA 121,0,144,248,201,
     44,240
239 DATA 186,208,152,165,53,
     133,99
240 DATA 165,54,133,98,76,
     142,166
241 DATA 165,99,24,101,49,
     133,99
242 DATA 165,98,101,50,133,
     98,32
243 DATA 201,198,208,251,96,
     32,2
244 DATA 198,32,201,198,32,
     201,198
245 DATA 208,8,169,255,133,
     99,133
246 DATA 98,48,14,32,201,198
     ,197
247 DATA 20,208,16,32,201,
     198,197
248 DATA 21,208,12,162,144,
     56,32
249 DATA 73,188,76,223,189,
     32,201
250 DATA 198,32,13,198,240,
     209,32
251 DATA 114,198,230,251,32
     ,165,198
252 DATA 230,45,208,2,230,46
     ,96
253 DATA 32,114,198,198,251,
     32,141
254 DATA 198,165,45,208,2,
     198,46
255 DATA 198,45,96,32,124,
     198,160
256 DATA 0,132,17,132,251,96
     ,165
257 DATA 122,133,34,165,123,
     133,35
258 DATA 165,45,133,36,165,
     46,133
259 DATA 37,96,164,17,200,
     177,34
260 DATA 164,251,200,145,34,
     32,190
261 DATA 198,208,1,96,230,
     34,208
262 DATA 236,230,35,208,232,
     164,17
263 DATA 177,36,164,251,145,
     36,32
264 DATA 190,198,208,1,96,
     165,36
265 DATA 208,2,198,37,198,36
     ,76
266 DATA 165,198,165,34,197,
     36,208
267 DATA 4,165,35,197,37,96,
     160
268 DATA 0,230,122,208,2,230
     ,123
269 DATA 177,122,96,32,51,
     165,165
270 DATA 34,166,35,24,105,2,
     133
271 DATA 45,144,1,232,134,
     46,32
272 DATA 89,166,76,116,164,
     0,137
273 DATA 138,141,167,-1
```

Si en lugar de utilizar el cargador BASIC prefieres recurrir a un ensamblador, aquí tienes el listado.

```
DIR.  HEXADEC. NEMONICO
===== ======== =========

,C000 20 9E B7 JSR $B79E
,C003 8E 86 02 STX $0286
,C006 20 FD AE JSR $AEFD
,C009 20 9E B7 JSR $B79E
,C00C 8A       TXA
,C00D 48       PHA
,C00E 20 FD AE JSR $AEFD
,C011 20 9E B7 JSR $B79E
,C014 68       PLA
,C015 A8       TAY
,C016 18       CLC
,C017 20 F0 FF JSR $FFF0
,C01A 20 FD AE JSR $AEFD
,C01D 20 A4 AA JSR $AAA4
,C020 60       RTS
,C021 20 9E B7 JSR $B79E
,C024 8E 86 02 STX $0286
,C027 20 FD AE JSR $AEFD
,C02A 20 9E B7 JSR $B79E
,C02D 8E 20 D0 STX $D020
,C030 20 FD AE JSR $AEFD
,C033 20 9E B7 JSR $B79E
,C036 8E 21 D0 STX $D021
,C039 60       RTS
,C03A 20 F7 B7 JSR $B7F7
,C03D 20 FD AE JSR $AEFD
,C040 20 EB B7 JSR $B7EB
,C043 8E 13 03 STX $0313
,C046 A9 00    LDA #$00
,C048 85 02    STA $02
,C04A 20 FD AE JSR $AEFD
,C04D 20 9E B7 JSR $B79E
,C050 8A       TXA
,C051 A4 02    LDY $02
,C053 91 14    STA ($14),Y
,C055 CC 13 03 CPY $0313
,C058 F0 05    BEQ $C05F
,C05A E6 02    INC $02
,C05C 4C 4A C0 JMP $C04A
,C05F 60       RTS
,C060 20 8A AD JSR $AD8A
,C063 20 F7 B7 JSR $B7F7
,C066 A5 14    LDA $14
,C068 85 FB    STA $FB
,C06A A5 15    LDA $15
,C06C 85 FC    STA $FC
,C06E 20 FD AE JSR $AEFD
,C071 20 8A AD JSR $AD8A
,C074 20 F7 B7 JSR $B7F7
,C077 20 FD AE JSR $AEFD
,C07A 20 9E B7 JSR $B79E
,C07D 86 02    STX $02
,C07F A5 15    LDA $15
,C081 C5 FC    CMP $FC
,C083 90 23    BCC $C0A8
,C085 D0 06    BNE $C08D
,C087 A5 14    LDA $14
,C089 C5 FB    CMP $FB
,C08B 90 1B    BCC $C0A8
,C08D A5 02    LDA $02
,C08F A0 00    LDY #$00
,C091 91 FB    STA ($FB),Y
,C093 A5 FB    LDA $FB
,C095 C5 14    CMP $14
,C097 D0 06    BNE $C09F
,C099 A5 FC    LDA $FC
,C09B C5 15    CMP $15
,C09D F0 09    BEQ $C0A8
,C09F E6 FB    INC $FB
,C0A1 D0 EA    BNE $C08D
,C0A3 E6 FC    INC $FC
,C0A5 4C 8D C0 JMP $C08D
,C0A8 60       RTS
,C0A9 20 8A AD JSR $AD8A
,C0AC 20 F7 B7 JSR $B7F7
```

```
,C0AF A5 14    LDA $14
,C0B1 85 FB    STA $FB
,C0B3 A5 15    LDA $15
,C0B5 85 FC    STA $FC
,C0B7 20 FD AE JSR $AEFD
,C0BA 20 8A AD JSR $AD8A
,C0BD 20 F7 B7 JSR $B7F7
,C0C0 A5 14    LDA $14
,C0C2 85 FD    STA $FD
,C0C4 A5 15    LDA $15
,C0C6 85 FE    STA $FE
,C0C8 20 FD AE JSR $AEFD
,C0CB 20 8A AD JSR $AD8A
,C0CE 20 F7 B7 JSR $B7F7
,C0D1 A5 FE    LDA $FE
,C0D3 C5 FC    CMP $FC
,C0D5 90 29    BCC $C100
,C0D7 D0 06    BNE $C0DF
,C0D9 A5 FD    LDA $FD
,C0DB C5 FB    CMP $FB
,C0DD 90 21    BCC $C100
,C0DF A0 00    LDY #$00
,C0E1 B1 FB    LDA ($FB),Y
,C0E3 91 14    STA ($14),Y
,C0E5 A5 FB    LDA $FB
,C0E7 C5 FD    CMP $FD
,C0E9 D0 06    BNE $C0F1
,C0EB A5 FC    LDA $FC
,C0ED C5 FE    CMP $FE
,C0EF F0 0F    BEQ $C100
,C0F1 E6 FB    INC $FB
,C0F3 D0 02    BNE $C0F7
,C0F5 E6 FC    INC $FC
,C0F7 E6 14    INC $14
,C0F9 D0 E4    BNE $C0DF
,C0FB E6 15    INC $15
,C0FD 4C DF C0 JMP $C0DF
,C100 60       RTS
,C101 20 8A AD JSR $AD8A
,C104 20 F7 B7 JSR $B7F7
,C107 A5 14    LDA $14
,C109 85 FB    STA $FB
,C10B A5 15    LDA $15
,C10D 85 FC    STA $FC
,C10F 20 FD AE JSR $AEFD
,C112 20 8A AD JSR $AD8A
,C115 20 F7 B7 JSR $B7F7
,C118 A0 00    LDY #$00
,C11A A5 14    LDA $14
,C11C 91 FB    STA ($FB),Y
,C11E A5 15    LDA $15
,C120 C8       INY
,C121 91 FB    STA ($FB),Y
,C123 60       RTS
,C124 20 9E B7 JSR $B79E
,C127 20 FF E9 JSR $E9FF
,C12A 60       RTS
,C12B 20 8A AD JSR $AD8A
,C12E 20 F7 B7 JSR $B7F7
,C131 A0 00    LDY #$00
,C133 B1 14    LDA ($14),Y
,C135 AA       TAX
,C136 C8       INY
,C137 B1 14    LDA ($14),Y
,C139 20 CD BD JSR $BDCD
,C13C 60       RTS
,C13D 20 8A AD JSR $AD8A
,C140 20 F7 B7 JSR $B7F7
,C143 A5 14    LDA $14
,C145 85 FB    STA $FB
,C147 A5 15    LDA $15
,C149 85 FC    STA $FC
,C14B 20 FD AE JSR $AEFD
,C14E 20 8A AD JSR $AD8A
,C151 20 F7 B7 JSR $B7F7
```

```
,C154 A5 15     LDA $15
,C156 C5 FC     CMP $FC
,C158 90 3D     BCC $C197
,C15A D0 06     BNE $C162
,C15C A5 14     LDA $14
,C15E C5 FB     CMP $FB
,C160 90 35     BCC $C197
,C162 A9 00     LDA #$00
,C164 85 FD     STA $FD
,C166 85 FE     STA $FE
,C168 18        CLC
,C169 A0 00     LDY #$00
,C16B B1 FB     LDA ($FB),Y
,C16D AA        TAX
,C16E 65 FD     ADC $FD
,C170 85 FD     STA $FD
,C172 A5 FE     LDA $FE
,C174 69 00     ADC #$00
,C176 85 FE     STA $FE
,C178 A9 00     LDA #$00
,C17A 20 CD BD  JSR $BDCD
,C17D A9 2C     LDA #$2C
,C17F 20 D2 FF  JSR $FFD2
,C182 A5 FB     LDA $FB
,C184 C5 14     CMP $14
,C186 D0 06     BNE $C18E
,C188 A5 FC     LDA $FC
,C18A C5 15     CMP $15
,C18C F0 09     BEQ $C197
,C18E E6 FB     INC $FB
,C190 D0 D6     BNE $C168
,C192 E6 FC     INC $FC
,C194 4C 68 C1  JMP $C168
,C197 A9 23     LDA #$23
,C199 20 D2 FF  JSR $FFD2
,C19C A6 FD     LDX $FD
,C19E A5 FE     LDA $FE
,C1A0 4C CD BD  JMP $BDCD
,C1A3 20 8A AD  JSR $AD8A
,C1A6 20 F7 B7  JSR $B7F7
,C1A9 A2 00     LDX #$00
,C1AB E8        INX
,C1AC D0 FD     BNE $C1AB
,C1AE C6 14     DEC $14
,C1B0 A9 FF     LDA #$FF
,C1B2 C5 14     CMP $14
,C1B4 D0 F5     BNE $C1AB
,C1B6 C6 15     DEC $15
,C1B8 C5 15     CMP $15
,C1BA D0 EF     BNE $C1AB
,C1BC 60        RTS
,C1BD 20 9E B7  JSR $B79E
,C1C0 A0 00     LDY #$00
,C1C2 E0 01     CPX #$01
,C1C4 D0 02     BNE $C1C8
,C1C6 A0 07     LDY #$07
,C1C8 E0 02     CPX #$02
,C1CA D0 02     BNE $C1CE
,C1CC A0 0E     LDY #$0E
,C1CE 84 02     STY $02
,C1D0 20 FD AE  JSR $AEFD
,C1D3 20 9E B7  JSR $B79E
,C1D6 8E 18 D4  STX $D418
,C1D9 20 FD AE  JSR $AEFD
,C1DC 20 9E B7  JSR $B79E
,C1DF 8A        TXA
,C1E0 A4 02     LDY $02
,C1E2 99 05 D4  STA $D405,Y
,C1E5 20 FD AE  JSR $AEFD
,C1E8 20 9E B7  JSR $B79E
,C1EB 8A        TXA
,C1EC A4 02     LDY $02
,C1EE 99 06 D4  STA $D406,Y
,C1F1 20 FD AE  JSR $AEFD
,C1F4 20 9E B7  JSR $B79E
,C1F7 8A        TXA
,C1F8 A4 02     LDY $02
,C1FA 99 04 D4  STA $D404,Y
,C1FD 8E 13 03  STX $0313
,C200 20 FD AE  JSR $AEFD
,C203 20 9E B7  JSR $B79E
,C206 8A        TXA
,C207 A4 02     LDY $02
,C209 99 01 D4  STA $D401,Y
,C20C 20 FD AE  JSR $AEFD
,C20F 20 9E B7  JSR $B79E
,C212 8A        TXA
,C213 A4 02     LDY $02
,C215 99 00 D4  STA $D400,Y
,C218 CE 13 03  DEC $0313
,C21B AD 13 03  LDA $0313
,C21E A4 02     LDY $02
,C220 99 04 D4  STA $D404,Y
,C223 60        RTS
,C224 A2 00     LDX #$00
,C226 8A        TXA
,C227 9D 00 D4  STA $D400,X
,C22A E8        INX
,C22B E0 19     CPX #$19
,C22D D0 F8     BNE $C227
,C22F 60        RTS
,C230 4C 44 E5  JMP $E544
,C233 4C 18 E5  JMP $E518
,C236 4C EA E8  JMP $E8EA
,C239 A9 00     LDA #$00
,C23B 8D 8A 02  STA $028A
,C23E 60        RTS
,C23F A9 80     LDA #$80
,C241 8D 8A 02  STA $028A
,C244 60        RTS
,C245 A9 00     LDA #$00
,C247 85 C6     STA $C6
,C249 60        RTS
,C24A A9 ED     LDA #$ED
,C24C 8D 28 03  STA $0328
,C24F 60        RTS
,C250 A9 FB     LDA #$FB
,C252 8D 28 03  STA $0328
,C255 60        RTS
,C256 4C 66 E5  JMP $E566
,C259 A9 1B     LDA #$1B
,C25B 8D 11 D0  STA $D011
,C25E 60        RTS
,C25F A9 0B     LDA #$0B
,C261 8D 11 D0  STA $D011
,C264 60        RTS
,C265 A9 15     LDA #$15
,C267 8D 18 D0  STA $D018
,C26A 60        RTS
,C26B A9 17     LDA #$17
,C26D 8D 18 D0  STA $D018
,C270 60        RTS
,C271 A9 09     LDA #$09
,C273 4C D2 FF  JMP $FFD2
,C276 A9 08     LDA #$08
,C278 4C D2 FF  JMP $FFD2
,C27B 20 8A AD  JSR $AD8A
,C27E 20 F7 B7  JSR $B7F7
,C281 4C A3 A8  JMP $A8A3
,C284 20 8A AD  JSR $AD8A
,C287 20 F7 B7  JSR $B7F7
,C28A A9 00     LDA #$00
,C28C A8        TAY
,C28D 91 14     STA ($14),Y
,C28F 18        CLC
,C290 A5 14     LDA $14
,C292 69 01     ADC #$01
,C294 85 2B     STA $2B
,C296 A5 15     LDA $15
,C298 69 00     ADC #$00
,C29A 85 2C     STA $2C
,C29C 4C 9A E3  JMP $E39A
,C29F A9 3E     LDA #$3E
,C2A1 20 D2 FF  JSR $FFD2
,C2A4 A9 12     LDA #$12
,C2A6 20 D2 FF  JSR $FFD2
,C2A9 A5 37     LDA $37
,C2AB 38        SEC
,C2AC E5 2D     SBC $2D
,C2AE AA        TAX
,C2AF A5 38     LDA $38
,C2B1 E5 2E     SBC $2E
,C2B3 20 CD BD  JSR $BDCD
,C2B6 A9 60     LDA #$60
,C2B8 A0 E4     LDY #$E4
```

```
,C2BA 4C 1E AB JMP $AB1E
,C2BD A9 00    LDA #$00
,C2BF 85 C6    STA $C6
,C2C1 A5 C6    LDA $C6
,C2C3 C9 01    CMP #$01
,C2C5 D0 FA    BNE $C2C1
,C2C7 4C 92 AB JMP $AB92
,C2CA A9 08    LDA #$08
,C2CC A0 01    LDY #$01
,C2CE 91 2B    STA ($2B),Y
,C2D0 20 33 A5 JSR $A533
,C2D3 18       CLC
,C2D4 A5 22    LDA $22
,C2D6 69 02    ADC #$02
,C2D8 85 2D    STA $2D
,C2DA 85 2F    STA $2F
,C2DC 85 31    STA $31
,C2DE A5 23    LDA $23
,C2E0 69 00    ADC #$00
,C2E2 85 2E    STA $2E
,C2E4 85 30    STA $30
,C2E6 85 32    STA $32
,C2E8 60       RTS
,C2E9 20 8A AD JSR $AD8A
,C2EC 20 F7 B7 JSR $B7F7
,C2EF A5 14    LDA $14
,C2F1 85 3F    STA $3F
,C2F3 A5 15    LDA $15
,C2F5 85 40    STA $40
,C2F7 20 13 A6 JSR $A613
,C2FA 38       SEC
,C2FB A5 5F    LDA $5F
,C2FD E9 01    SBC #$01
,C2FF 85 41    STA $41
,C301 A5 60    LDA $60
,C303 E9 00    SBC #$00
,C305 85 42    STA $42
,C307 60       RTS
,C308 A2 00    LDX #$00
,C30A B5 2B    LDA $2B,X
,C30C 95 97    STA $97,X
,C30E E8       INX
,C30F E0 04    CPX #$04
,C311 D0 F7    BNE $C30A
,C313 20 8A AD JSR $AD8A
,C316 20 F7 B7 JSR $B7F7
,C319 A5 14    LDA $14
,C31B 85 2B    STA $2B
,C31D A5 15    LDA $15
,C31F 85 2C    STA $2C
,C321 20 FD AE JSR $AEFD
,C324 20 8A AD JSR $AD8A
,C327 20 F7 B7 JSR $B7F7
,C32A A5 14    LDA $14
,C32C 85 2D    STA $2D
,C32E A5 15    LDA $15
,C330 85 2E    STA $2E
,C332 20 56 E1 JSR $E156
,C335 A2 00    LDX #$00
,C337 B5 FB    LDA $FB,X
,C339 95 2B    STA $2B,X
,C33B E8       INX
,C33C E0 04    CPX #$04
,C33E D0 F7    BNE $C337
,C340 60       RTS
,C341 20 9E B7 JSR $B79E
,C344 86 02    STX $02
,C346 20 FD AE JSR $AEFD
,C349 20 9E B7 JSR $B79E
,C34C 8E 13 03 STX $0313
,C34F A5 02    LDA $02
,C351 C9 00    CMP #$00
,C353 F0 0B    BEQ $C360
,C355 AD 13 03 LDA $0313
,C358 20 D2 FF JSR $FFD2
,C35B C6 02    DEC $02
,C35D 4C 4F C3 JMP $C34F
,C360 60       RTS
,C361 4C 9A E3 JMP $E39A
,C364 20 8A AD JSR $AD8A
,C367 20 F7 B7 JSR $B7F7
,C36A AA       TAX
,C36B A9 48    LDA #$48
,C36D 20 D2 FF JSR $FFD2
,C370 A9 27    LDA #$27
,C372 20 D2 FF JSR $FFD2
,C375 A9 24    LDA #$24
,C377 20 D2 FF JSR $FFD2
,C37A 8A       TXA
,C37B 20 8B C3 JSR $C38B
,C37E 8A       TXA
,C37F 20 90 C3 JSR $C390
,C382 98       TYA
,C383 20 8B C3 JSR $C38B
,C386 98       TYA
,C387 20 90 C3 JSR $C390
,C38A 60       RTS
,C38B 18       CLC
,C38C 6A       ROR
,C38D 6A       ROR
,C38E 6A       ROR
,C38F 6A       ROR
,C390 29 0F    AND #$0F
,C392 18       CLC
,C393 69 30    ADC #$30
,C395 C9 3A    CMP #$3A
,C397 90 02    BCC $C39B
,C399 69 06    ADC #$06
,C39B 20 D2 FF JSR $FFD2
,C39E 60       RTS
,C39F A9 44    LDA #$44
,C3A1 20 D2 FF JSR $FFD2
,C3A4 A9 27    LDA #$27
,C3A6 20 D2 FF JSR $FFD2
,C3A9 20 BA C3 JSR $C3BA
,C3AC 85 22    STA $22
,C3AE 20 BA C3 JSR $C3BA
,C3B1 AA       TAX
,C3B2 A5 22    LDA $22
,C3B4 20 CD BD JSR $BDCD
,C3B7 4C E4 A7 JMP $A7E4
,C3BA 20 CB C3 JSR $C3CB
,C3BD 0A       ASL
,C3BE 0A       ASL
,C3BF 0A       ASL
,C3C0 0A       ASL
,C3C1 85 23    STA $23
,C3C3 20 CB C3 JSR $C3CB
,C3C6 65 23    ADC $23
,C3C8 85 23    STA $23
,C3CA 60       RTS
,C3CB 20 73 00 JSR $0073
,C3CE C9 3A    CMP #$3A
,C3D0 29 0F    AND #$0F
,C3D2 90 02    BCC $C3D6
,C3D4 69 08    ADC #$08
,C3D6 60       RTS
,C3D7 20 8A AD JSR $AD8A
,C3DA 20 F7 B7 JSR $B7F7
,C3DD A9 5B    LDA #$5B
,C3DF 20 D2 FF JSR $FFD2
,C3E2 A9 00    LDA #$00
,C3E4 A6 14    LDX $14
,C3E6 20 CD BD JSR $BDCD
,C3E9 A9 2C    LDA #$2C
,C3EB 20 D2 FF JSR $FFD2
,C3EE A9 00    LDA #$00
,C3F0 A6 15    LDX $15
,C3F2 20 CD BD JSR $BDCD
,C3F5 A9 5D    LDA #$5D
,C3F7 20 D2 FF JSR $FFD2
,C3FA 60       RTS
,C3FB 20 9E B7 JSR $B79E
,C3FE 86 02    STX $02
,C400 20 FD AE JSR $AEFD
,C403 20 9E B7 JSR $B79E
,C406 8A       TXA
,C407 A6 02    LDX $02
,C409 4C CD BD JMP $BDCD
,C40C 05 00    ORA $00
,C40E 00       BRK
,C40F 20 79 00 JSR $0079
,C412 D0 06    BNE $C41A
,C414 A9 00    LDA #$00
,C416 8D 0E C4 STA $C40E
```

```
,C419 60        RTS
,C41A A9 01     LDA #$01
,C41C 8D 0E C4  STA $C40E
,C41F A9 35     LDA #$35
,C421 8D 04 03  STA $0304
,C424 A9 C4     LDA #$C4
,C426 8D 05 03  STA $0305
,C429 20 8A AD  JSR $AD8A
,C42C 20 F7 B7  JSR $B7F7
,C42F A5 14     LDA $14
,C431 8D 0C C4  STA $C40C
,C434 A5 15     LDA $15
,C436 8D 0D C4  STA $C40D
,C439 60        RTS
,C43A AD 00 02  LDA $0200
,C43D C9 30     CMP #$30
,C43F 90 3B     BCC $C47C
,C441 C9 3A     CMP #$3A
,C443 B0 37     BCS $C47C
,C445 AD 0E C4  LDA $C40E
,C448 F0 32     BEQ $C47C
,C44A 20 7C A5  JSR $A57C
,C44D 84 02     STY $02
,C44F AD 0C C4  LDA $C40C
,C452 18        CLC
,C453 65 14     ADC $14
,C455 85 63     STA $63
,C457 AD 0D C4  LDA $C40D
,C45A 65 15     ADC $15
,C45C 85 62     STA $62
,C45E A2 90     LDX #$90
,C460 38        SEC
,C461 20 49 BC  JSR $BC49
,C464 20 DF BD  JSR $BDDF
,C467 85 FE     STA $FE
,C469 84 F4     STY $F4
,C46B A0 00     LDY #$00
,C46D B1 FE     LDA ($FE),Y
,C46F F0 06     BEQ $C477
,C471 99 77 02  STA $0277,Y
,C474 C8        INY
,C475 D0 F6     BNE $C46D
,C477 84 C6     STY $C6
,C479 A4 02     LDY $02
,C47B 60        RTS
,C47C 4C 7C A5  JMP $A57C
```

....

```
,CD00 A9 0B     LDA #$0B
,CD02 8D 08 03  STA $0308
,CD05 A9 CD     LDA #$CD
,CD07 8D 09 03  STA $0309
,CD0A 60        RTS
,CD0B 20 73 00  JSR $0073
,CD0E C9 40     CMP #$40
,CD10 F0 03     BEQ $CD15
,CD12 4C E7 A7  JMP $A7E7
,CD15 A0 01     LDY #$01
,CD17 B1 7A     LDA ($7A),Y
,CD19 85 FF     STA $FF
,CD1B A0 02     LDY #$02
,CD1D B1 7A     LDA ($7A),Y
,CD1F 85 02     STA $02
,CD21 A2 00     LDX #$00
,CD23 BD 80 CE  LDA $CE80,X
,CD26 C5 FF     CMP $FF
,CD28 D0 09     BNE $CD33
,CD2A E8        INX
,CD2B BD 80 CE  LDA $CE80,X
,CD2E C5 02     CMP $02
,CD30 F0 10     BEQ $CD42
,CD32 CA        DEX
,CD33 C9 00     CMP #$00
,CD35 F0 06     BEQ $CD3D
,CD37 E8        INX
,CD38 E8        INX
,CD39 E0 80     CPX #$80
,CD3B D0 E6     BNE $CD23
,CD3D A2 0B     LDX #$0B
,CD3F 6C 00 03  JMP ($0300)
,CD42 86 02     STX $02
,CD44 20 73 00  JSR $0073
,CD47 20 73 00  JSR $0073
,CD4A 20 73 00  JSR $0073
,CD4D A6 02     LDX $02
,CD4F BD FF CD  LDA $CDFF,X
,CD52 85 FC     STA $FC
,CD54 BD 00 CE  LDA $CE00,X
,CD57 85 FD     STA $FD
,CD59 A9 4C     LDA #$4C
,CD5B 85 FB     STA $FB
,CD5D 20 FB 00  JSR $00FB
,CD60 4C AE A7  JMP $A7AE
,CD63 60        RTS
```

....

```
,CE00 00        BRK
,CE01 C0 21     CPY #$21
,CE03 C0 40     CPY #$40
,CE05 C0 60     CPY #$60
,CE07 C0 A9     CPY #$A9
,CE09 C0 01     CPY #$01
,CE0B C1 24     CMP ($24,X)
,CE0D C1 2B     CMP ($2B,X)
,CE0F C1 3D     CMP ($3D,X)
,CE11 C1 A3     CMP ($A3,X)
,CE13 C1 BD     CMP ($BD,X)
,CE15 C1 24     CMP ($24,X)
,CE17 C2        ???
,CE18 30 C2     BMI $CDDC
,CE1A 33        ???
,CE1B C2        ???
,CE1C 36 C2     ROL $C2,X
,CE1E 39 C2 3F  AND $3FC2,Y
,CE21 C2        ???
,CE22 45 C2     EOR $C2
,CE24 4A        LSR
,CE25 C2        ???
,CE26 50 C2     BVC $CDEA
,CE28 56 C2     LSR $C2,X
,CE2A 59 C2 5F  EOR $5FC2,Y
,CE2D C2        ???
,CE2E 65 C2     ADC $C2
,CE30 6B        ???
,CE31 C2        ???
,CE32 71 C2     ADC ($C2),Y
,CE34 76 C2     ROR $C2,X
,CE36 7B        ???
,CE37 C2        ???
,CE38 84 C2     STY $C2
,CE3A 9F        ???
,CE3B C2        ???
,CE3C BD C2 CA  LDA $CAC2,X
,CE3F C2        ???
,CE40 E9 C2     SBC #$C2
,CE42 08        PHP
,CE43 C3        ???
,CE44 41 C3     EOR ($C3,X)
,CE46 61 C3     ADC ($C3,X)
,CE48 64        ???
,CE49 C3        ???
,CE4A 9F        ???
,CE4B C3        ???
,CE4C D7        ???
,CE4D C3        ???
,CE4E FB        ???
,CE4F C3        ???
,CE50 0F        ???
,CE51 C4 49     CPY $49
,CE53 C5 00     CMP $00
```

....

Listado de bytes
(no es programa)

```
,CE80 99 25 43  STA $4325,Y
,CE83 B0 97     BCS $CE1C
,CE85 52        ???
,CE86 97        ???
,CE87 53        ???
,CE88 4D 89 50  EOR $5089
,CE8B 97        ???
,CE8C 9C        ???
,CE8D 27        ???
,CE8E 50 C2     BVC $CE52
,CE90 4D 87 92  EOR $9287
,CE93 27        ???
```

```
,CE94 53        ???
,CE95 91 53     STA ($53),Y
,CE97 8B        ???
,CE98 53        ???
,CE99 9C        ???
,CE9A 53        ???
,CE9B A2 55     LDX #$55
,CE9D 50 A5     BVC $CE44
,CE9F 30 A5     BMI $CE46
,CEA1 31 4B     AND ($4B),Y
,CEA3 9C        ???
,CEA4 96 30     STX $30,Y
,CEA6 96 31     STX $31,Y
,CEA8 A4 50     LDY $50
,CEAA 9E        ???
,CEAB 30 9E     BMI $CE4B
,CEAD 31 C6     AND ($C6),Y
,CEAF 30 C6     BMI $CE77
,CEB1 31 91     AND ($91),Y
,CEB3 30 91     BMI $CE46
,CEB5 31 47     AND ($47),Y
,CEB7 89        ???
,CEB8 42        ???
,CEB9 A2 4D     LDX #$4D
,CEBB B8        CLV
,CEBC 92        ???
,CEBD A1 A1     LDA ($A1,X)
,CEBF A2 4C     LDX #$4C
,CEC1 8C 4D 94 STY $944D
,CEC4 43        ???
,CEC5 C7        ???
,CEC6 4E A2 45 LSR $45A2
,CEC9 27        ???
,CECA 48        PHA
,CECB 27        ???
,CECC B9 54 BE
LDA $BE54,Y
,CECF 54        ???
,CED0 91 A4
STA ($A4),Y
```



BUSCA TU NOMBRE

UN CICLOMOTOR VESPINO

* ANTONIO FERNANDEZ DELGADO, ALCALA DE HENARES (MADRID).

UNA CADENA HIFI DE SPECTRAVIDEO

* JUAN LOPEZ SALA, ALICANTE.
* EMILIO DE ANDRES DE LA VEGA, MADRID.
* J. LORENZO LASO SOLANA, R. DE CAMARGO (CANTABRIA).

UN LP DE GABINETE CALIGARI

* SANTIAGO CAMPAYO CUESTA, BADALONA (BARCELONA).
* ALBERTO VILARMAU ESPINAL (BARCELONA).
* LUIS ALBERTO CARRION MORENO, VALLADOLID.
* MANUEL CASTILLEJOS VAYA, MADRID.
* MANUEL ALONSO MARTINEZ, BILBAO.
* PEDRO BARAHONA CORDERO, COSLADA (MADRID).
* MIGUEL SUNE RODRIGUEZ. BARCELONA.
* JOSECHU FERNANDEZ BAREA, SANTURCE (VIZCAYA).
* JOSE LUIS MARTINEZ MUNOZ, STA. COLOMA DE GRAMANET (BARCELONA).
* GONZALO NOVOA SOTELO, ORENSE.

UN LOTE DE SOFTWARE DE ERBE

* SANTIAGO ESTEFANIA BARRIO, GETXO (VIZCAYA).
* J. MANUEL COLOMINA MESTRES, STA. COLOMA DE GRAMANET (BARCELONA).
* VICTORINO RODRIGUEZ VALDESOGO, LEON.
* J. MANUEL PEREIRA GARCIA, BARCELONA.
* MA. ANGUSTIAS DELGADO, GRANADA.
* MANUEL TERRER MENA, CARTAGENA (MURCIA).
* MIGUEL SALAS GARCIA, MADRID.
* J. CARLOS DE LA LLANA SANTO, REUS (TARRAGONA).
* SERGIO MARTINEZ COLLDEFORN, BARCELONA.
* ROBERTO TORRIJOS SANTOS, TALAVERA (TOLEDO).

TRES PROGRAMAS DE COMPULOGICAL

* FERNANDO GONZALEZ RODRIGUEZ, LA OROTAVA (TENERIFE).
* J. ANTONIO POZO GALLETERO, CHIVA (VALENCIA).
* A. MANUEL YANEZ VALEIRO, BILBAO.
* JORDI MARZO VIDAL, BARCELONA.
* JOAN CARRASCO DOMINGUEZ, TERRASA (BARCELONA).
* ANSELMO FERNANDEZ BERODAS, MALAGA.
* FRANCISCO MACIA LACUSA, BARCELONA.
* ANDRES ORTIZ MENDEZ, MARTOREL (BARCELONA).
* ESTEBAN HORCAJO ARROYO, S.S. DE LOS REYES (MADRID).
* JAVIER SUBIRATS CERVERA, REUS (TARRAGONA).

TRES PROGRAMAS DE PROEINSA

* JORDI PRUNES CASTERAS, TERRASA (BARCELONA).
* DAVID BRAVO CALVO, BARCELONA.
* ANTONIO GRIMALT ANDREU, MANACOR (MALLORCA).
* RUBEN DOMINGUEZ LOPEZ, AGUADULE (ALMERIA).
* SALVADOR DE LA FUENTE FERNANDEZ, VALENCIA.
* ALFONSO MANTECON LAGE, LAS ROZAS (MADRID).
* MIGUEL LA TORRE ROVIROSA, VILAFRANCA DEL PENEDES (BARCELONA).
* M. ANGEL FLORES ACUNA, LA CORUNA.
* F. CARLOS GAITAN ALFARO, CORDOBA.
* J. MIGUEL TORES CARO, CADIZ.

TRES PROGRAMAS DE MICROUNO

* MANUEL PINEIRO BESADA, PORRINO (PONTEVEDRA).
* A. JOSE PALACIOS SOLER, VALENCIA.
* SEBASTIAN MONZON PEREZ, SEVILLA.
* J. CARLOS CAPAFONS CUBERO, SITGES (BARCELONA).
* JAVIER HINOJOSA RUIZ, CASTELLAR DEL VALLES (BARCELONA).
* ALFREDO CANTELI SUAREZ, OVIEDO (ASTURIAS).
* R. ALBERTO GARGALLO CASCAN, BARCELONA.
* SANTIAGO FERREIRO PARAMO, LA CORUNA.
* ANTONIO GALAN ELEZ, MADRID.
* JESUS VALERO ALCARAZ, MADRID.

CUATRO PROGRAMAS DE SERMA

* ANTONIO GOMEZ AGUILAR, MALAGA.
* PAQUI RODRIGUEZ GONZALEZ, BADAJOZ.
* FERNANDO ROJAS PALACIOS, ALGECIRAS (CADIZ).
* RAFAEL ASENJO PLAZA, MARACENA (GRANADA).
* J. MANUEL PARRA BARBERO, LA HERRADURA (GRANADA).
* PABLO HERRERA PEREZ GRUESO, MORA (TOLEDO).
* TEOFILO DEL RINCON GONZALEZ, PUIG DEN VALS (IBIZA).
* JOSE LUIS GONZALEZ OLEA, FUENLABRADA (MADRID).
* JAIME SOLA FLOX, VILANOVA I LA GELTRUB' (BARCELONA).
* ALVARO DAPENA URCOLA, ZARAGOZA.

UN KIT ROBOTICO DE FERRE-MORET

* JOSE JULIO VELAZQUEZ, LEJONA (VIZCAYA).

UN JOYSTICK DE COMPULOGICAL

* MIGUEL MILLAN DE LA TORRE, S.JUAN DESPI (BARCELONA).
* VICENTE MORALEDA HIDALGO, SEVILLA.
* RAMON JUNQUERA SOBRON, GALDAKANO (VIZCAYA).

UN JOYSTICK DE MICROUNO

* JOSE C. RODRIGUEZ ESCOBAR, MADRID.
* BENITO SANTAMARIA ROMERO, MADRID.
* JOSE ANTONIO GONZALEZ CUETO, GIJON (ASTURIAS).
* DANIEL ROSS ESTRAGUEZ, BARCELONA.
* ANTONIO DIAZ GONZALEZ, SANTANDER.
* EDUARDO TORRECILLAS VALVERDE, BADALONA (BARCELONA).
* PEDRO LANCILLA TELLA, ZARAGOZA.
* V. ANTONIO UCEDA ALVAREZ, GETAFE (MADRID).
* J. MANUEL VIGO COUSINO, VIGO (PONTEVEDRA).
* MARIANO LOPEZ LOPEZ, COLMENAR VIEJO (MADRID).
* JAVIER CORONA DELGADO, PAMPLONA.
* FCO. JAVIER DIAZ GARZON, BARCELONA.
* J. MANUEL TRIAS COMPANY, SOLLER (MALLORCA).
* PABLO VAQUERO SEPULVEDA, AVILA.
* JESUS OCANA GOMEZ, MADRID.

UN CRONOMETRO DE MICROUNO

* JUAN PENA SANCHEZ, ISLA CRISTINA (HUELVA).
* ANGEL GOMEZ LOPEZ, LUGO.
* IGNACIO TIERNO MARTINEZ, BARCELONA.
* MA. LUISA SANZ AGUADO, LLAVERNERS (BARCELONA).
* J. LUIS BLASCO CABAL, MADRID.

UN LIBRO DE ANAYA MULTIMEDIA

* ANTONIO CABEZA HERNANDEZ, TALAVERA (TOLEDO).
* JAVIER MARTINEZ CORMENEZANA, BILBAO.
* MIGUEL SERRANO DEL MAZO, CACERES.
* ECKART SCHILDER, BREXO (LA CORUNA).
* SANTIAGO DIAZ PEREZ, PORTUGALETE (VIZCAYA).
* RICARDO SEMPERE PALAU, QUART DE POBLET (VALENCIA).
* FRANCISCO TINEO RAMIREZ, MATARO (BARCELONA).
* MIGUEL ANGEL FERNANDEZ, MADRID.
* JORGE ABAD GIL, MADRID.
* EVA GOMEZ RODRIGO, SALAMANCA.
* F. JAVIER VAZQUEZ SERRANO, CORDOBA.
* J. CARLOS GARRIDO BERNARDEZ, ORENSE.
* J. MIGUEL MELLADO BREY, SAN ROQUE (CADIZ).
* SANTIAGO HERNANDEZ SANCHEZ, FUENLABRADA (MADRID).
* GINES CLIMENT PEREZ, VILLAJOYOSA (ALICANTE).

UN LAPIZ OPTICO DE COMPULOGICAL

* G. IVAN GALLARDO REYES, BADALONA (BARCELONA).
* JOSE JULIO NAVARRO GARCIA, MURCIA.
* JOQUIN GARCIA SANCHEZ, BEJAR (SALAMANCA).

UN TECLADO MUSIC MAKER DE MICROELECTRONICA Y CONTROL

* LUIS MARIA CORTEGOSO LOBATO, VIGO (PONTEVEDRA).
* L. MIGUEL PRIETO ACEVEDO, BILBAO.
* ESTEBAN GIL BASALO, HUELVA.
* VICENTE DIAZ FERNANDEZ, MADRID.
* L. GONZALO VILA QUINTANS, LA CORUNA.
* NICOLAS CHACON SANTIAGO, CORDOBA.
* PABLO ALEX JIMENEZ, GRANADA.
* MIGUEL BURGOS GRACIA, SEGOVIA.
* JOAN SIEIRA RIBOT, HOSTAL RICH (SEGOVIA).

UN DIGITALIZADOR DE VIDEO DE COMPULOGICAL

* PILAR SOTO GAVALDA, TARRAGONA.
* L. CARLOS MATINEZ ZAPATA, MADRID.
* VICENTE TORRES JIMENEZ, GRANADA.

UN SINTETIZADOR DE VOZ DE MICROELECTRONICA Y CONTROL

* RICARDO GONZALEZ RODRIGUEZ, GUADALAJARA.
* JOSE CRUZ GONZALEZ, BARCELONA.
* FRANCISCO SANCHEZ DE LA MAZA, CORDOBA.
* ANTON SANROMA GARIT, VALLS (BARCELONA).
* LUIS CORTES NUNEZ, BADALONA (BARCELONA).
* F. MANUEL REINA SANCHEZ, UVA DEL ROSARIO (MALAGA).
* JORDI BENET GILI, TARREGA (LERIDA).
* LUIS MIGUEL DIEZ TASCON, OVIEDO.
* CARLOS TEMES CASAS, BASTAGUEIRO (LA CORUNA).

SIMULADOR DE VUELO

- LA SIMULACION DE VUELO
- PROGRAMAS DE ENTRENAMIENTO
- VUELO MEDIANTE INSTRUMENTOS
- EL MOVIMIENTO DEL AVION
- PERDIDA DE VELOCIDAD

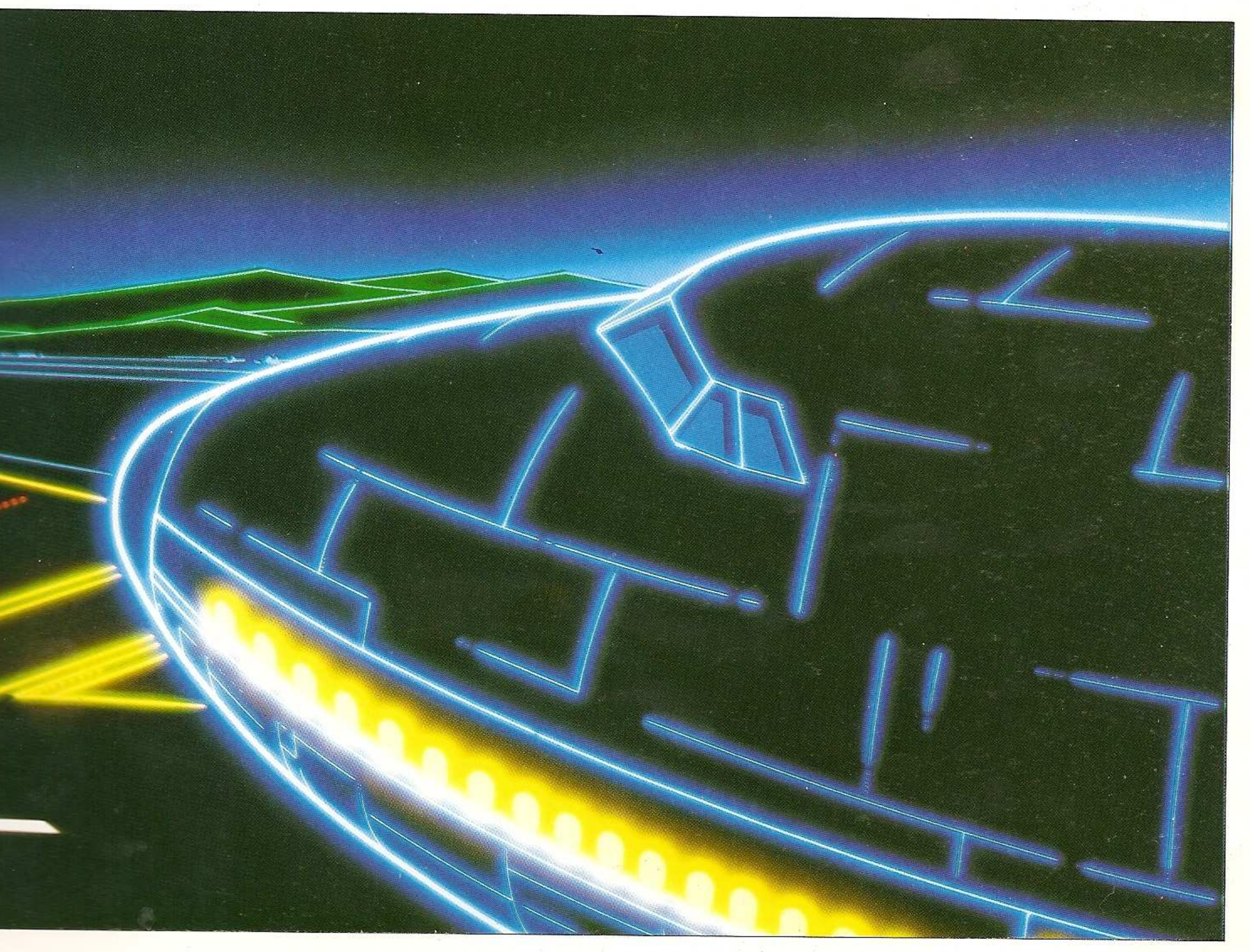
Este programa de simulación de vuelo es similar a los que se emplean en las escuelas de vuelo para enseñar a los pilotos cómo tienen que volar utilizando únicamente sus instrumentos; en la primera parte se reproduce la cabina

Los programas de juegos varían desde una fantasía desbordante que supone la entrada a mundos imaginarios y la participación en aventuras, hasta la simulación de situaciones de la vida real. Esto te permite poner a prueba tu capacidad en situaciones potencialmente peligrosas, sin tener que hacerte daño o perder totalmente millones de pesetas en costosos equipos.

Los programas de simulación de vuelo contienen un elemento de fantasía: tú solo en la cabina, con toda la tripulación aquejada de una misteriosa enfermedad, con una sola mano consigues hacer que el avión tome tierra felizmente. Pero los sofisticados programas de este tipo tienen un uso práctico real, hasta el punto de que casi todas las principales compañías de líneas aéreas y escuelas de vuelo los utilizan con regularidad.

SIMULADORES DE ENTRENAMIENTO

En el extremo superior de la escala está la simulación total, la llamada «Fase 3» en la terminología de las administraciones de aviación civil, que te permite experimentar las mismas sensaciones que un piloto en un avión de verdad. Tú ves lo mismo que él ve a



través de la ventanilla de la carlinga (incluyendo una pequeña diferencia de ángulo en el punto de vista para la posición del copiloto); sentirás lo mismo que él siente en los despegues y en los aterrizajes, así como las turbulencias; oirás lo mismo que él oye, incluyendo las indicaciones del control de tráfico aéreo. En teoría un piloto puede completar todo su entrenamiento en uno de estos simuladores y obtener su licencia sin tener que abandonar el suelo para nada.

SIMULADORES DE SOBREMESA

En el extremo opuesto de la escala están los programas de simulación de vuelo muy parecidos al que veremos a continuación.

Las unidades de sobremesa se pueden «volar» en el interior de un aula, y resultan útiles para la enseñanza de los procedimientos de cabina y para desarrollar la rapidez de reflejos de los pilotos.

Resultan además esenciales para la enseñanza del vuelo por instrumentos, una técnica que permite al piloto navegar apoyándose únicamente en el panel de instrumentos, algo que todo piloto ha de hacer cuando las condiciones meteorológicas son malas.

LO QUE HACE EL PROGRAMA

Este artículo consta de tres partes y en él se presenta un programa de simulación de vuelo en el que se supone que te has hecho cargo del control del avión cuando éste se encuentra a 2000 metros de altura y a 20000 metros de distancia de la pista de aterrizaje que constituye el objetivo. Por la ventanilla de la cabina es muy poco lo que puedes ver, solamente el horizonte, cuando hay visibilidad, y un punto distante que es la pista de aterrizaje, por lo cual como piloto sensato que eres, tendrás que confiar en tu experiencia y atendiendo a lo que te indique el panel de instrumentos ponerte a salvo en tierra a tí y a tus pasajeros.

LOS INSTRUMENTOS

En el panel de instrumentos hay cuatro *diales*. El primero te informa de la velocidad del aire. Este valor cambia según que estés picando (tu velocidad aumenta), elevándote (tu velocidad disminuye) o modificando la potencia de los motores. Un contador que hay debajo del *dial* de la velocidad del aire te indica la marcación de tu brújula.

El segundo *dial* te muestra dónde está el horizonte en relación con el aeroplano. Esto significa que aunque el horizonte no resulte visible a través de la ventanilla de la carlinga, todavía sigues sabiendo dónde está. El contador que hay debajo de este *dial* te indica la marcación de la pista de aterrizaje.

El tercer *dial* te proporciona una lectura de la altitud. Tiene dos manecillas, una para los miles y otra para los cientos. El contador que hay debajo calcula la deriva del avión; como la pista de aterrizaje tiene una anchura de 100 metros una deriva de +50 o −50 hará que la pierdas.

El último *dial* te indica la velocidad del motor en revoluciones por minuto. El contador que hay debajo te permite conocer la distancia a la que te encuentras del centro de la pista.

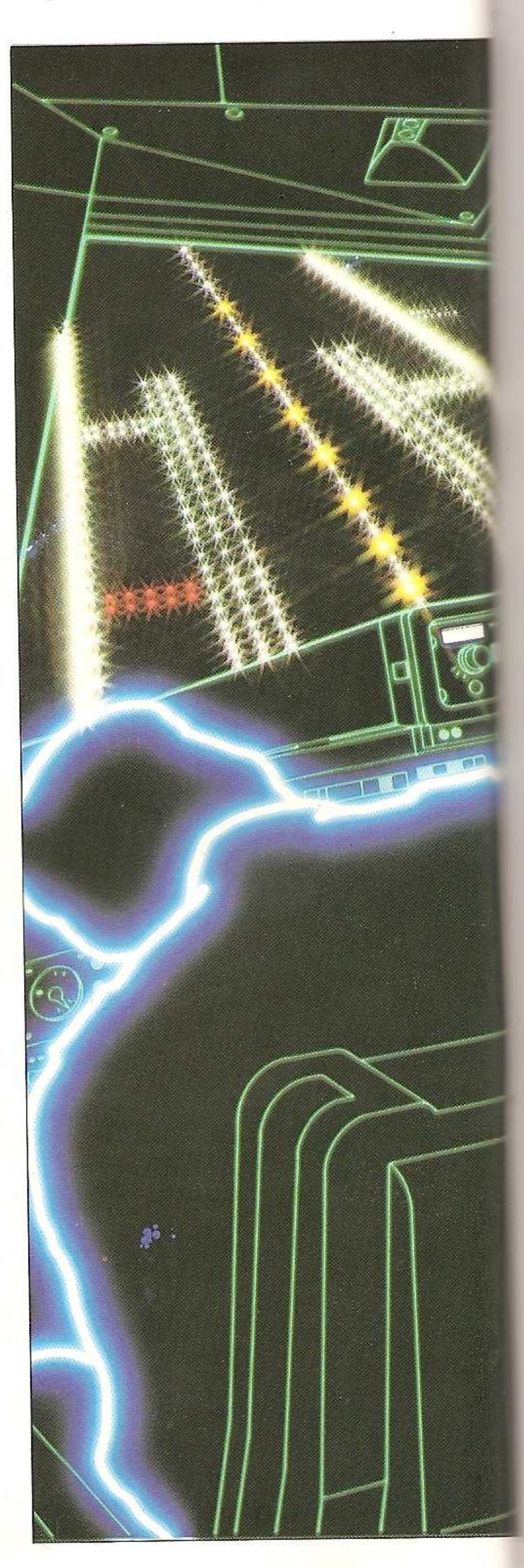
ATERRIZAJE

Aunque hay programas de simuladores de vuelo en los que la imagen que se ve a través de la ventanilla de la cabina se va haciendo más nítida a medida que progresa la aproximación, no es éste nuestro caso. Por ello debes centrar la imagen radar de la pista de aterrizaje.

En el momento en que asumes el control, las condiciones meteorológicas son buenas y la pista se encuentra en la dirección norte. El aterrizaje en estas condiciones no es difícil y el juego perdería en seguida su atractivo si no se pudieran modificar estas condiciones. Para añadir más dificultad, puedes especificar la velocidad y la dirección del viento: por ejemplo un furioso vendaval que sople de lado te pondrá las cosas mucho más difíciles.

MOVIENDO EL AVION

El margen de los controles de que dispones se aproxima bastante a los

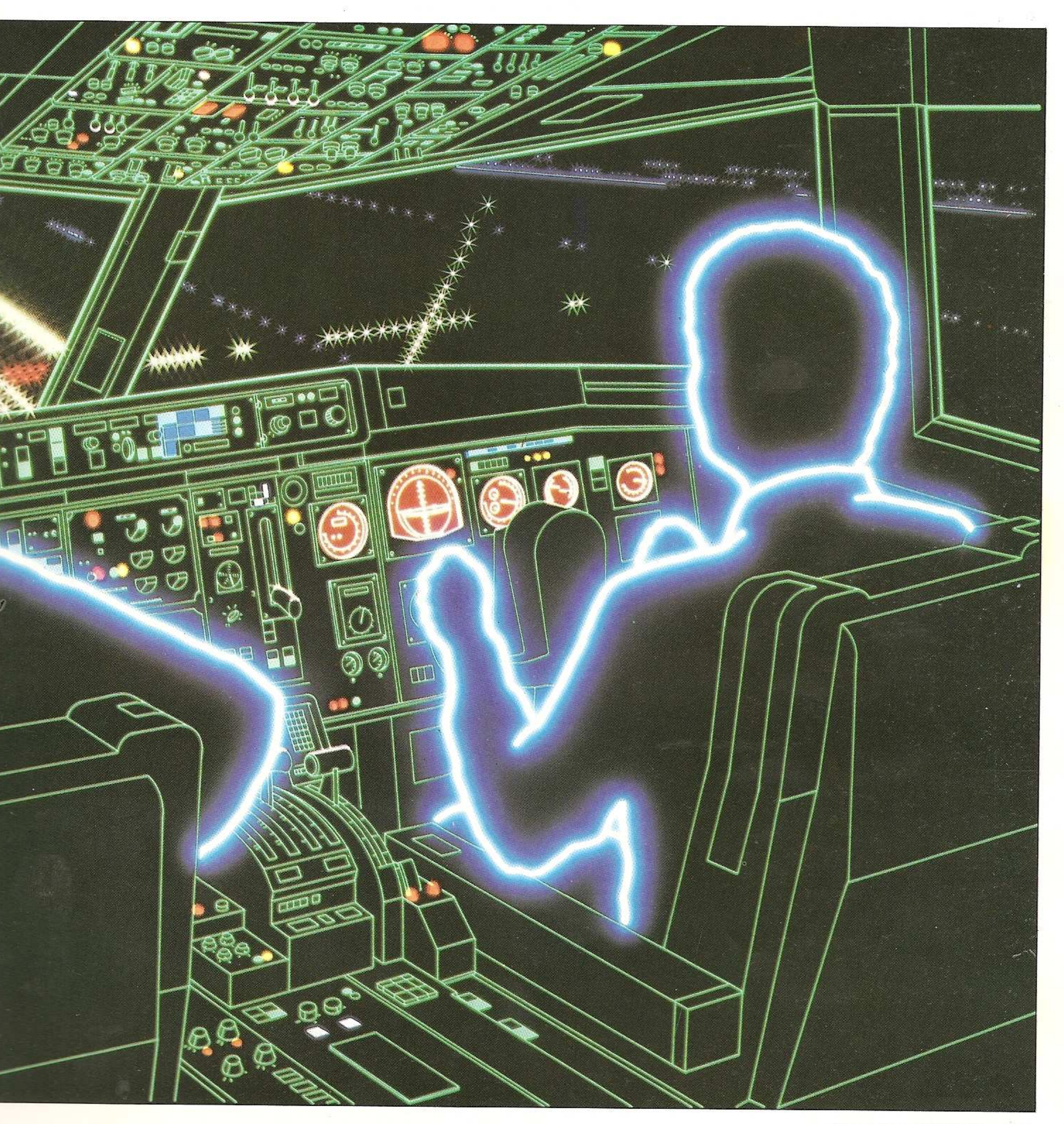


controles de un avión de verdad, aunque estés pulsando teclas en vez de utilizar un *joystick*.

En una aeronave real, el control de la elevación —los movimientos hacia arriba y hacia abajo— se ejerce moviendo el *joystick* hacia atrás o adelante, lo que hace que se muevan los elevadores del plano de cola hacia arriba o abajo. Tú vas a utilizar dos teclas para conseguir este mismo efecto; en la tercera sección de este artículo teclearás la parte de programa que se ocupa de esto.

El balance del aeroplano —el movimiento lateral— se controla moviendo el *joystick* de un lado a otro. Esto hace que se muevan los alerones, que son las superficies de control que hay en las alas. También en este caso utilizarás dos teclas para hacer que se muevan hacia la derecha o hacia la izquierda.

Tus dos últimos controles te permi-



ten acelerar o retardar la velocidad del motor, lo cual resulta esencial para controlar los tiempos en la maniobra de aterrizaje y para asegurarte de que no entrarás en pérdida de velocidad.

PERDIDA DE VELOCIDAD

Cuando la velocidad de un avión cae por debajo de un determinado valor, se dice que entra en pérdida de velocidad y en ese momento el avión empieza a caer como si fuera una piedra. En este programa, cuando la velocidad de tu avión cae por debajo de los 30 metros por segundo, el aeroplano caerá en picado abruptamente. Si tenías bastante altura en el momento en que esto ocurre, una acción rápida te puede salvar, pero una entrada en pérdida es algo que aterroriza a cualquier piloto.

DIVISION DEL PROGRAMA

El programa es demasiado largo y complejo como para darlo todo de una sola vez, por lo que lo hemos dividido en tres partes.

Lo que se hace en esta primera parte es configurar la pantalla para mostrar el interior de la cabina, con su ventanilla, los cuatro *diales* debidamente etiquetados y los letreros de los contadores.

Los comandos que intervienen serán familiares para la mayoría de vosotros por haberlos visto ya en otros programas.

La parte de programa introducida en la parte dos, hace posible que los *diales* y contadores sean sensibles al movimiento del avión y hay un comando temporal que hace que éste vuele aleatoriamente sin que haya un piloto que actúe sobre los controles, para que puedas ver funcionando el panel de instrumentos. La sección final te permite tomar el control del avión y realizar una estimación de tu técnica de aterrizaje para que puedas juzgar tus progresos.

DIBUJANDO LA CABINA

Teclea en tu ordenador el siguiente programa para tener dibujada la cabina

Para poder acceder a los gráficos de alta resolución del **Commodore 64**, hemos escrito este programa con el Simon's BASIC. Esto significa que no puedes hacerlo correr en un **Commodore** normal sin conectar el cartucho. No obstante, los que no tengáis este cartucho no os desesperéis. Pronto veremos en INPUT un programa en código máquina que os permitirá ejecutar estos programas. La única pequeña modificación que tendrás que hacer es poner una @ delante de cada comando de gráficos.

Teclea para Commodore-64

```
1 REM**ES NECESARIO EL
  CARTUCHO SIMON'S BASIC
2 REM**[PI]=TECLAS[SHIFT+
  ^]
5100 A$="VEL.AIRE HORIZ.
     ALTITUD[][][]RPM"
5110 B$="CABECEO PISTA
     DERIVA DISTANCIA"
5120 HIRES 0,1:MULTI 4,0,5:
     COLOUR 0,1
5130 BLOCK 0,110,160,200,2
5140 TEXT 0,120,A$,3,1,5:
     TEXT 0,175,B$,1,1,5
5150 LINE 0,171,160,171,0:
     LINE 0,200,160,200,0
5160 FOR Z=0 TO 3:CIRCLE 20+
     Z*40,150,15,15,0
5170 IF Z=1 THEN TEXT 57,146,
     "[COMM.+R]",0,1,1:
     NEXT Z
5180 FOR K=0 TO 9:PLOT(20+Z*
     40)+17*SIN(K*[PI]/5),
     150-19*COS(K*[PI]/5),0:
     NEXT K,Z
5190 GOTO 5190
```

A$ y B$ definidos en las líneas 5100 y 5110 contienen las etiquetas de los *diales* y los contadores. La línea 130 dibuja un bloque negro debajo de la ventanilla de la cabina, para que contraste el interior oscuro del avión con el azul claro de la ventanilla. La línea 5140 sitúa los letreros y la línea 5150 dibuja dos líneas para sugerir que los contadores etiquetados están dispuestos en un panel separado de los *diales*. La línea 5160 dibuja los círculos de los *diales* (realmente son algo elípticos, debido a la forma de trabajar de la pantalla gráfica) y el diagrama del avión en el *dial* del horizonte.

La línea 5180 dibuja los indicadores que rodean a los otros tres *diales*.

Hasta aquí, este programa te mostrará una consola de vuelo simulada. En la próxima parte podrás teclear la sección de programa que hará posible que tu avión vuele.



DESPEGA PARA TU PRIMER VUELO

- VUELO CON EL PILOTO AUTOMATICO
- APROXIMACION A LA PISTA DE ATERRIZAJE
- DIBUJANDO LA TRAYECTORIA
- EL PANEL DE INSTRUMENTOS

En la segunda parte del simulador de vuelo, puedes arrancar los motores y obsevar cómo adquiere vida tu panel de instrumentos. Pero ten cuidado: ¡El piloto automático se ha vuelto loco!

En la primera parte has tecleado las líneas que reproducían sobre la pantalla el interior de la cabina de vuelo.

En esta parte verás adquirir movimiento a tu avión y cómo adquiere vida el panel de instrumentos; de modo que, aunque todavía no has tomado los mandos, puedes ver cómo responde el panel de instrumentos del avión ante los movimientos del mismo.



EL AVION VUELA

Esta es la parte más larga del programa, con bastante diferencia. Hay una larga serie de variables interdependientes que tienen que ser constantemente actualizadas para controlar el avance del avión. Además hay que volver a dibujar constantemente el panel de instrumentos a medida que van cambiando la posición y la altura del avión, ya que los *diales* tienen que ir describiendo el movimiento.

LA APROXIMACION A LA PISTA

Hay una imagen radar de la pista de aterrizaje que te muestra el ángulo con el que te estás aproximando.

DIBUJO DE LA TRAYECTORIA

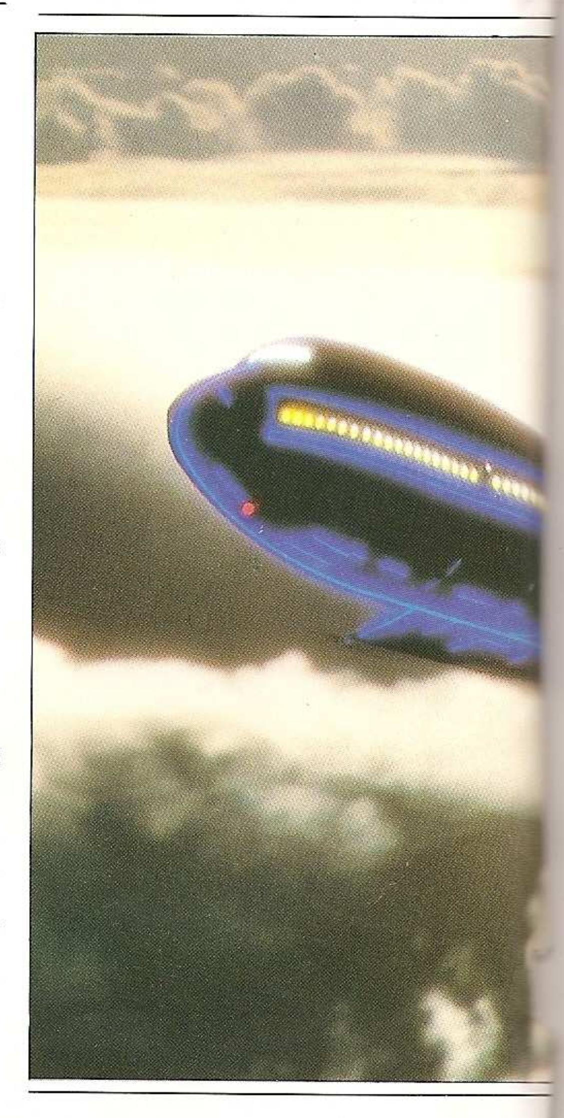
Para poder dibujar con precisión la posición del avión, hay que tener en cuenta muchos factores. Por ejemplo, la dirección en la que estás volando, se ve afectada por la dirección del viento y por el balance del avión. La velocidad de avance depende parcialmente de la velocidad del viento. La altura que vas perdiendo, o ganando, está relacionada con la velocidad de vuelo, etc.

Para poder actualizar los *diales* y los contadores, hay que hacer una estimación de las variables que van cambiando, con arreglo a la forma en que afectan a las lecturas, después de lo cual se puede hacer de nuevo el dibujo.

Teclea para Commodore-64

```
10 GOTO 5000
500 RA=AD*C:VX=AS*SIN(RA)
510 VY=AS*COS(RA):RETURN
1000 PZ=PZ+GZ:PY=PY+GY:PX=PX
     +GX
```

```
1010 PT=PT+NC:RL=RL+BC
1020 AS=AS+16*(TC*30-AS-8*PT
     )/AS
1030 IF SL=1 THEN TEXT 60,50,
     "PERDIDA",0,3,8:SL=0:
     GOTO 1050
1040 IF AS<30 THEN GOSUB
     1500
1050 AD=AD+RL:IF AD<0 THEN
     AD=AD+360
1060 IF AD>359 THEN AD=AD-
     360
1070 VZ=AS*SIN(PT*C)-10+AS
     /15
1080 GZ=VZ:GY=VY+WY:
     GX=VX+WX
1090 IF VY=0 THEN GD=-[PI]/2
     :GOTO 1110
1100 GD=-ATN(VX/VY)/C
1110 GOSUB 500
1120 RETURN
1500 SL=1:TEXT 60,50,
     "PERDIDA",1,3,8
1510 RL=INT(RND(1)*21)-9:PT=
     21-INT(RND(1)*5)
1520 RETURN
2000 LINE 20,150,20+13*SIN
     (AS*[PI]/200),
     150-13*
     COS(AS*[PI]/200)
     ,4
2010 TN=PZ/1000:UN=PZ-1000*
     INT(TN)
2020 LINE 100,150,100+6*SIN
     (TN*[PI]/5),150-6*COS
     (TN*[PI]/5),4
2030 LINE 100,150,100+13*SIN
     (UN*[PI]/500),
     150-13*
     COS(UN*[PI]/500)
     ,4
2040 LINE 140,150,140+13*SIN
     (TC*[PI]/5),150-13*COS
     (TC*[PI]/5),4
2050 TEXT 0,190,STR$(ABS(INT
     (AD))),4,1,7:RETURN
2060 IF PY=0 THEN RB=0
2065 IF PY<>0 THEN RB=ATN(PX
     /PY)/C:IF PY>0 THEN RB=
     RB+180
2070 IF RB<0 THEN RB=RB+360
2075 GOSUB 7000
2080 TEXT 35,190,STR$(INT(RB
     )),1,1,7:TEXT 70,190,
     STR$(ABS(INT(PX))),1,1,
     7
2090 TEXT 110,190,STR$(INT(
     SQR(PY*PY+PX*PX))),1,1,
     7
2095 S1=INT(RB):S2=ABS(INT
     (PX))
2096 S3=INT(SQR
     (PY*PY+PX*PX))
2098 IF KJ=1 THEN LINE X1,
     Y1,X2,Y2,4
2100 KJ=0:YC=150+(PT/3):X1=
     50:X2=70:Y1=YC+17*TAN(
     RL*2*C):Y2=YC-17*TAN(RL
     *2*C)
2110 IF(YC<137 OR YC>163)
     AND RL=0 THEN 2320
2120 IF Y1<137 THEN X1=60-(
     60-X1)*(140-YC)/
     (Y1-YC):Y1=140::GOTO
     2140
2130 IF Y1>163 THEN X1=60-(
     60-X1)*(160-YC)/(Y1-YC)
     :Y1=160
2140 IF Y2<137 THEN X2=60-(
     60-X2)*(140-YC)/
     (Y2-YC):Y2=140:GOTO
     2160
2150 IF Y2>163 THEN X2=60-(
     60-X2)*(160-YC)/(Y2-YC)
     :Y2=160
2160 IF X1<50 OR X2>70 THEN
     2190
2170 LINE X1,Y1,X2,Y2,
     4:KJ=1
2190 IF RL=RR AND PP=PT THEN
     2290
2200 IF HF=1 THEN LINE X3,
     Y3,X4,Y4,0
2210 HF=0:YC=33+PT*4:X3=0:
     X4=159:Y3=YC+59*TAN(RL*
     2*C):Y4=YC-59*TAN(RL*
     2*C)
2220 IF(YC<0 OR YC>109) AND
     RL=0 THEN 2290
2230 IF Y3<0 THEN X3=80-(80-
     X3)*(-YC)/(Y3-YC):Y3=0:
     GOTO 2250
2240 IF Y3>109 THEN X3=80-(
     80-X3)*(109-YC)/(Y3-YC)
     :Y3=109
2250 IF Y4<0 THEN X4=80-(80-
     X4)*(-YC)/(Y4-YC):Y4=0:
     GOTO 2270
2260 IF Y4>109 THEN X4=80-(
     80-X4)*(109-YC)/(Y4-YC)
     :Y4=109
2270 IF X3<0 OR X4>159 THEN
     2290
2280 HF=1:LINEr X3,Y3,X4,Y4,
     3
2290 WB=AD:IF AD>180 THEN WB
     =WB-360
2300 IF RB>180 THEN WB=WB+
     360-RB:GOTO 2310
2305 WB=WB-RB
2310 IF ABS(WB)>60 AND ABS
     (PY)>1000 THEN 2350
```



```
2320 AN=59/(60*SQR((X3-X4)*
     (X3-X4)+(Y3-Y4)*(Y3-
     Y4)))
2325 X5=(X3+X4)/2+SGN(X3-X4)
     +WB*AN*(X3+X4)
2330 Y5=(Y3+Y4)/2+2+WB*AN*(
     Y3-Y4)
2335 IF X5<0 OR X5>159 OR Y5
     <0 OR Y5>109 THEN 2350
2340 IF ABS(PY)<1000 THEN R=
     8-Y5/10:GOTO 2350
```

```
2345 R=4000/ABS(PY):IF R*10+
     Y5>80 THEN R=8-
     Y5/10
2350 GOSUB 8000
2370 RR=RL:PP=RT:RETURN
5000 PRINT"[SHIFT+CLR/HOME]
     [CTRL+4]":NRM:COLOUR 6,
     6:PP=-1:RR=-1
5010 C=[PI]/180:PY=-20000:PZ
     =2000:AS=150
5020 PRINT" ESCRIBE LA VELO
     CIDAD DEL VIENTO ([CTRL
     +6]1-50[CTRL+4]) M/S"
5025 PRINT"Y LA DIRECCION
     ([CTRL+6]0-359[CTRL+4])
     GRADOS[CRSR abajo]":
     FLASH 5,10
5030 INPUT X0,X1:IF X0>50 OR
     X0<1 OR X1<0 OR X1>359
     THEN 5000
5040 X0=X0/3:OFF:POKE 650,
     128
5050 PRINT AT(0,20)"VEL.
     VIENTO =";3*X0;"M/S":
     PRINT" DIRECCION =";X1;
     "GRADOS"
5060 WY=-X0*COS(X1*C)
5070 WX=-X0*SIN(X1*C)
5080 GZ=VZ:GY=VY+WY:GX=
     VX+WX
5090 TC=5:RT=3:TP=5:WR=50:
     PAUSE 2
5190 REM
5500 GOSUB 2000:IF INT(RND(1
     )*10)=1 THEN RL=RL+SGN
     (TL)*INT(RND(1)*4)-1
5510 IF INT(RND(1)*10)=1
     THEN PT=PT+3-INT(RND(1)
     *4+1)*2
5520 GOSUB 1000:IF PZ<=0
     THEN 5540
5530 GOSUB 2000:GOSUB 2060:
     GOTO 5500
5540 GOTO 5540
7000 TEXT 35,190,STR$(S1),2,
     1,7:TEXT 70,190,STR$
     (S2),2,1,7
7010 TEXT 110,190,STR$(S3),
     2,1,7:RETURN
8000 IF WQ=1 THEN LINE 78,0,
     OX,OY,0
8010 WQ=1:OX=78-(16*SIN(RB*
     (PI/180))):OY=(16*ABS(
     COS(RB*([PI]/180))))
8020 LINE 78,0,OX,OY,2
8025 WB=AD:IF AD>180 THEN WB
     =WB-360
8026 IF RB>180 THEN WB=WB+
     360-RB:GOTO 8040
8030 WB=WB-RB
8040 IF RW=1 THEN PLOT G1,
     G2,0
8050 RW=0:IF ABS(WB)>57 THEN
     RETURN
8060 RX=X3+INT(((X4-X3)/2)
     -SIN(WB*([PI]/180))*(X4
     -X3)*.6)
8070 RY=Y3+((Y4-Y3)*((RX-X3)
     /(X4-X3))+2)
8080 IF RY<0 OR RY>109 OR RX
     <0 OR RX>159 THEN
     RETURN
8090 RW=1:PLOT RX,RY,2:G1=RX
     :G2=RY
8100 RETURN
```



En cuanto teclees RUN, el programa salta a la línea 5000. Las líneas 5000 y 5010 inicializan un grupo de variables que controlan la posición del avión en el aire. Inicialmente el avión está situado a 20000 metros de la pista de aterrizaje y a 2000 metros del suelo.

Las líneas 5020 a 5030 permiten al piloto elegir la intensidad y dirección del viento y comprueban que las entradas suministradas están dentro de los márgenes permitidos. Estos valores se imprimen en la línea 5050, antes de que las líneas 5060 y 5070 calculen las componentes de la velocidad del viento en la dirección de la marcha y en la dirección transversal a la misma. A continuación se utilizan WX y WY en la línea 5080 para ajustar la posición del avión.

La línea 5090 asigna los valores del cuentarrevoluciones (TC), los límites del balance (RT) y cabezada (TP) y la

anchura de la pista de aterrizaje (WR).

La línea 5500 llama a la subrutina que comienza en la línea 2000 y termina en la 2050. Esta subrutina actualiza los *diales*. Las líneas 2000, 2020, 2030 y 2040 dibujan las agujas con arreglo a la posición y velocidad.

Después del retorno a la línea 5500, el resto de la misma y toda la línea 5510 actúan supliendo a la rutina de control que añadirás en la próxima parte de este artículo.

La línea 5520 llama a la subrutina que empieza en la línea 1000. Esta subrutina actualiza todas las variables durante el vuelo del avión. Las líneas 1030 y 1040 comprueban si ha habido pérdida de velocidad, lo que ocurrirá si has permitido que la velocidad de tu avión caiga por debajo de los 30 metros por segundo. Caso de que ocurra así, el programa salta a la subrutina de la línea 1500, que presenta en pantalla el mensaje de la pérdida de velocidad y asigna un valor aleatorio al balance y la cabezada para simular la pérdida de control del aeroplano. La línea 1110 llama a la subrutina de la línea 500. Las líneas 500 y 510 calculan la velocidad de ascenso o descenso y la velocidad transversal.

Al volver el programa desde la línea 1120 a la línea 5520, comprueba si el avión ha tocado el suelo, saltando a la línea 5540 caso de que así sea. La línea 5540 es una detención temporal.

La línea 5530 llama a la subrutina que comienza en la línea 2000, que actualiza los *diales*. Seguidamente se llama a la subrutina de la línea 2060. Las líneas 2060 a 2070 actualizan la demora de la pista de aterrizaje antes de que se supriman las tres últimas presentaciones numéricas que figuran por acción de las líneas 2000 y siguientes. Las nuevas lecturas se presentan en las líneas 2080 y 2090. En las 2095 y 2096 se calculan nuevos valores para S1, S2 y S3; valores que se utilizan en las líneas 7000 y 7010.

El horizonte artificial se dibuja por medio de la sentencia LINE que figura en la línea 2098; la correspondiente sentencia de borrado está en la línea 2170, donde KJ es simplemente un indicador para que el ordenador sepa cuándo tiene que dibujar un nuevo horizonte artificial. Las líneas 2100 a 2160 actualizan las variables de control. Las líneas 2200 a 2280 se ocupan del horizonte real.



En las líneas 2290 a 2370 se calcula la nueva posición de la pista de aterrizaje, utilizando la subrutina de las líneas 8000 a 8100 para dibujar la posición adecuada.

Una vez que el horizonte ha sido dibujado de nuevo, la presentación se actualiza completamente y la línea 5530 envía el programa a la 5500. Observa que el bucle principal del programa son precisamente estas cuatro líneas.



Instrúcciones y Pantallas
en **CASTELLANO**
Commodore 64
pvp 2500. pts

MAIL ORDER MONSTERS
Monstruos por Correo s.l.

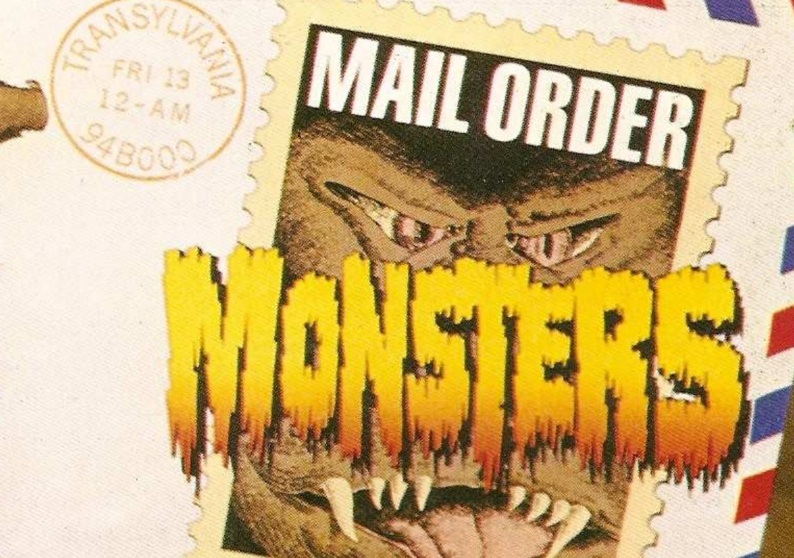
Bienvenido a "Monstruos por correo S.L.", la mejor compañía de monstruos de la Galaxia.

Nuestros esmerados servicios te ofrecen los más poderosos monstruos a medida.

Por solo 500 psychons podrás organizarte las combinaciones de monstruos y armas que siempre habías deseado para arrasar en los torneos de tu planeta.

Paul Reiche III, Evan Robinson, Nicky Robinson
ELECTRONIC ARTS
2155 Campus Drive
SAN MATEO, CA 94403. USA

SPOEDBESTELLING
EXPRÈS



03314

DRO SOFT
Fundadores, 3
28028 - MADRID

ELECTRONIC ARTS

Editado por



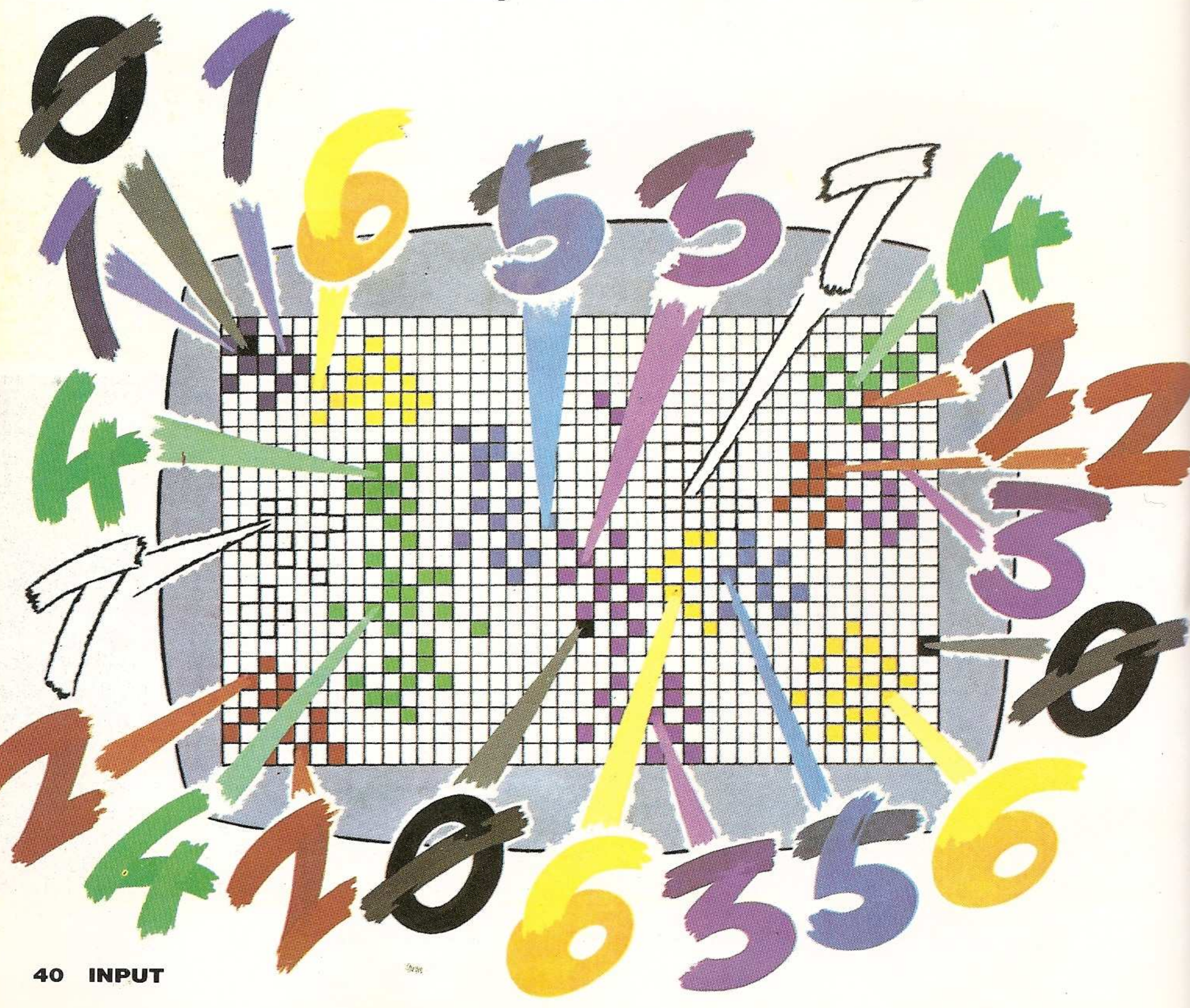
DETECCION DE OBJETOS EN LA PANTALLA

- DETECCION DE FORMAS EXTRAÑAS
- COMPROBACIONES DE COLOR
- EL COMANDO PEEK
- EL PROGRAMA DE LA PELOTA
- DETECCION DE LOS CHOQUES

Tu ordenador puede seguir complicados gráficos utilizando determinados comandos que le permiten observar su propia pantalla. Esto resulta especialmente útil en los juegos en los que se producen colisiones

En las situaciones en que hayas creado una figura detallada sobre la pantalla, ¿cómo puedes asegurarte de que el siguiente objeto que añadas a la imagen no coincidirá con algo que ya había allí antes?

Una respuesta obvia es que seas cuidadoso y sistemático al llevar la cuenta de las zonas de la pantalla que ya han sido utilizadas. Pero esto puede llegar a resultar enormemente complicado, y hay circunstancias en que puede ser casi imposible, como es el caso de los gráficos móviles. Y sin embargo éste es precisamente el tipo de problemas con el que te encuentras al escribir programas de juegos y cosas parecidas en las que hay invasores y vehículos espaciales moviéndose por una galaxia, o fantasmas recorriendo un laberinto. En tales casos tienes que asegurarte de que no se impriman dos gráficos en el mismo sitio, o de qué es lo que sucederá si se produce la colisión, por ejemplo, que se desintegre la nave espacial.



DETECCION DE FORMAS EXTRAÑAS

Supongamos por ejemplo que deseas escribir un programa que haga aparecer en pantalla una pelota dando botes. En este caso la cosa es muy sencilla: como conoces las coordenadas de los cuatro lados de la pantalla, puedes incluir en tu programa cuatro condiciones IF ... THEN para comprobar si la pelota ha llegado o no a uno de los cuatro lados. Ello se hace comparando las coordenadas de la pelota con las coordenadas conocidas de los lados. Pero ¿qué sucede cuando lo que quieres comprobar es si la pelota ha golpeado contra un objeto de forma más extraña, por ejemplo circular?

Podrías utilizar el mismo método, disponiendo unas cuantas comprobaciones del tipo IF ... THEN que contengan los detalles de las coordenadas del círculo para ver en qué puntos golpea la pelota con los lados. Pero debido a que la forma curva es especialmente compleja, se necesitaría un número de comprobaciones muy grande. Sabes que el ordenador necesita mucho tiempo para ejecutar cada comprobación IF ... THEN. Un programa escrito de esta forma resultaría extremadamente lento, y el movimiento resultante sobre la pantalla se produciría a base de tirones o sacudidas.

Existe un límite para las velocidades que es posible obtener utilizando el BASIC. Pero tu ordenador tiene un comando que te permite detectar la presencia sobre la pantalla de cualquier objeto con mucha mayor rapidez que la comprobación de las coordenadas. Esto es posible aunque no se conozca su posición.

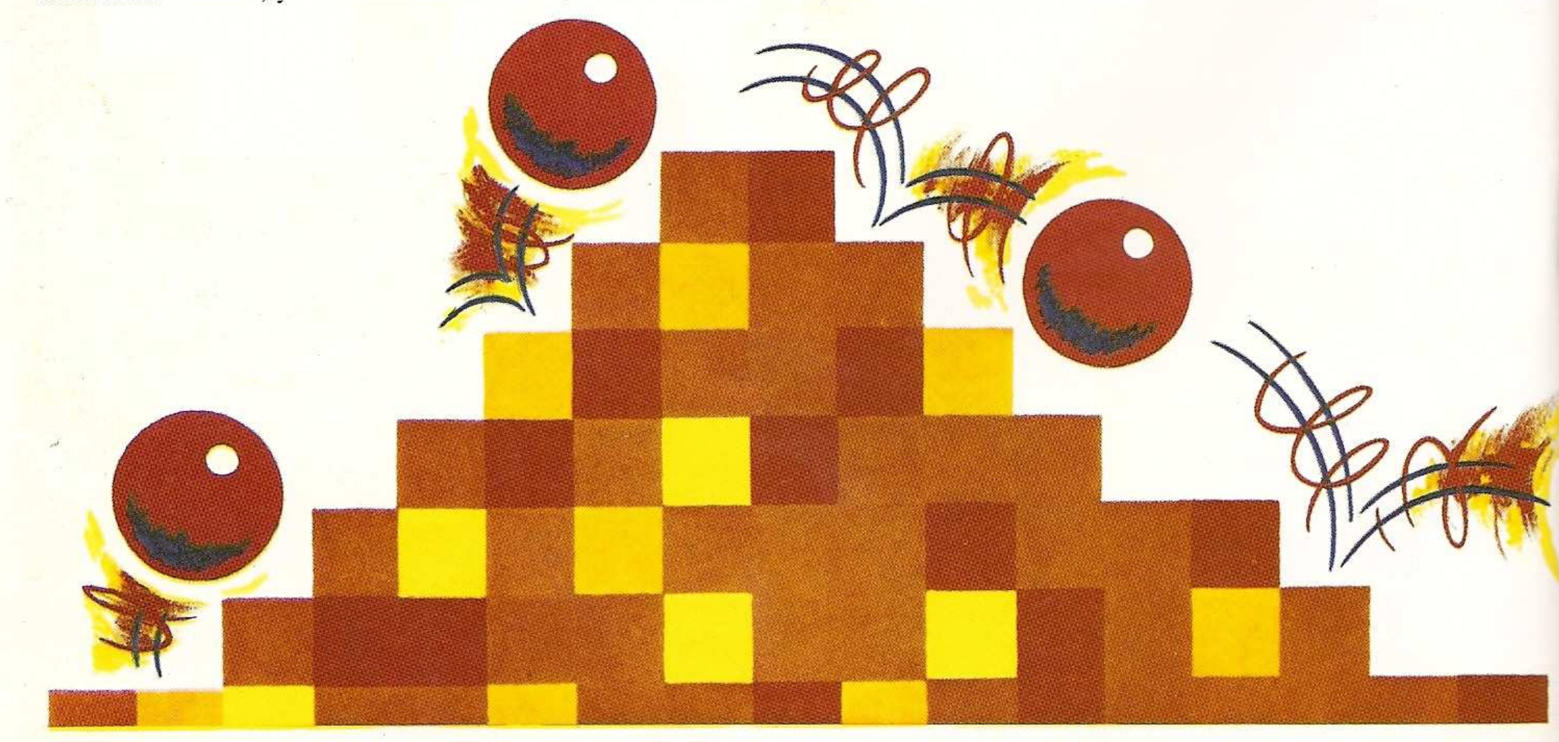
EL COLOR POR MEDIO DE NUMEROS

El comando PEEK en el **Commodore** te devuelve como respuesta el color especificado en cada carácter cuadrado. Esto significa que puedes detectar la presencia de cualquier objeto, sin más que especificar su color y comprobando todas las posiciones de pantalla. Así, en el ejemplo anterior, con un círculo rojo resultaría el número de código del color rojo cada vez que se presentara. Comprobando únicamente este código el programa podría hacer que la pelota rebotara hacia afuera del círculo, o cualquier otra cosa.

En realidad tanto el **Commodore 64** como el **Vic** dan como respuesta un número del 0 al 15, que es el número del color. El **Commodore 64** le suma 240 a cada número, con lo que resulta un número comprendido entre 240 y 255. No tienes más que restar 240 del número que obtengas, para tener el número del color. Así, si por ejemplo has hecho un PEEK en una determinada dirección de un carácter de color, y has obtenido como respuesta 240, el valor de color que resulta es el 0, que corresponde al negro.

En cuanto a la sintaxis de este comando, digamos que en el **Commodore** es ligeramente diferente de la utilizada en otros ordenadores, ya que en lugar de utilizar las coordenadas de pantalla situadas entre paréntesis para que el ordenador sepa en qué cuadrado estás interesado, lo que hace es aplicar un PEEK en las direcciones reales de memoria que contienen los detalles de la zona de pantalla de que se trate. La dirección del primer carácter cuadrado es 55296 y la del último carácter es 56295. Este «fichero» ocupa exactamente 1000 bytes; recuerda que la pantalla está formada por 25 líneas de 40 caracteres.

La primera dirección corresponde al carácter cuadrado situado en la esquina superior izquierda de la pantalla, la siguiente dirección es la del siguiente carácter de esa línea, etc. Esto significa que los detalles del color del carácter situado en la esquina inferior



derecha de la pantalla, están contenidos en la dirección 56295. En consecuencia el comando adopta la siguiente forma:

```
PRINT PEEK (56296) AND 15
```

(La última parte de la instrucción hace que resulte un número del 0 al 15). Por desgracia, no puedes invertir el proceso y cambiar los colores de cada carácter cuadrado. Sin embargo, sí que puedes POKEar diferentes valores en el fichero de atributos de tu ordenador.

Teclea y ejecuta el siguiente programa, que constituye un ejemplo de utilización del comando PEEK.

Teclea para C-64 y Vic-20

```
1 REM**[]SIGNIFICA ESPACIO
5 S=40:CC=55296:D=21:SC=1024
  :A1=18::A2=10
10 FOR Z=0 TO S-1:PRINT
   "[CTRL+5][SHIFT++]";:R$=
   R$+"[CRSR izq.]":S$=S$+
   "[]":NEXT Z
20 FOR Z=0 TO D:PRINT"
   [SHIFT++][CTRL+9][CRL+8]"
   LEFT$(S$,S-2)"[CTRL+5]
   CTRL+0][SHIFT++]";:D$=D$+
   "[CRSR abajo]":NEXT Z
30 FOR Z=0 TO S-1:PRINT
   "[SHIFT++]";:NEXT Z
50 FOR Z=1 TO S:X=(INT(RND
   (1)*A1)+1)*2:Y=(INT(RND
   (1)*A2)+1)*2
60 PRINT "[CLR/HOME][CRSR
   abajo]"LEFT$(D$,Y)LEFT$
   (R$,X)"[CTRL+9][]":NEXT
70 X=10:Y=10:XV=-1:YV=1
80 POKE SC+X+Y*S,215:OY=Y:
   OX=X
90 X=X+XV:Y=Y+YV
100 C1=PEEK(CC+(X-1)+Y*S)
    AND 15
105 C2=PEEK(CC+X+(Y-1)*S)
    AND 15
110 C3=PEEK(CC+(X+1)+Y*S)
    AND 15
115 C4=PEEK(CC+X+(Y+1)*S)
    AND 15
120 IF C1=4 OR C3=4 THEN XV=
    -XV
125 IF C2=4 OR C4=4 THEN YV=
    -YV
130 POKE SC+OX+OY*S,160:GOTO
    80
```

Para el **Vic**, teclea las líneas 10 a 130 igual que en el programa del **C-64**, añadiendo a continuación lo siguiente:

```
5 S=22:CC=38400:D=19:SC=7680
  :A1=9:A2=8
```

Los programas anteriores empiezan por establecer un contorno y una serie de cuadrados posicionados aleatoriamente, todos del mismo color (púrpura). La pelota es de un color distinto, y naturalmente lo mismo ocurre con el color de fondo.

La línea 5 define las variables que intervienen en el programa. S es el número de caracteres de una línea (se utiliza una variable para ello, ya que el **Commodore 64** y el **Vic** tienen pantallas de tamaños diferentes); CC es la dirección de comienzo de la memoria de color de la pantalla, que también es diferente para el **64** y el **Vic**, como también lo es la dirección de comienzo de la memoria de pantalla a la que hemos llamado SC. Las dos últimas variables, A1 y A2, son los límites entre los cuales el ordenador puede situar bloques aleatorios. Estos límites



se fijan para definir un área más pequeña que la definida por el contorno, para evitar que alguno de los bloques se sitúe en la misma posición que alguno de los de los bordes.

En las líneas 20 a 60 se establecen los bordes y se añaden los cuadrados aleatorios. A continuación el ordenador mueve su cursor y define cuatro nuevas variables (x e y para la posición de la pelota y xv e yv para la velocidad inicial en cada una de las dos direcciones). La línea 80 sirve para pokear la pelota en la pantalla. Esto es realmente lo que hace, más bien que imprimirla, ya que para hacer un PRINT se requerirían más caracteres de control del cursor, lo que haría que el programa resultase mucho más lento; en tal caso el ordenador tendría que desplazar primero el cursor y a continuación hacer un PRINT de la pelota.

Lo que realmente hace la instrucción de POKE es cargar el valor 215 en la posición de pantalla calculada por el programa, que es a la que debe ir a parar la pelota. El número 215 es el código ASCII del carácter que representa a la pelota.

En la línea 90 se suma la componente x de la velocidad (xv) a la coordenada x y la componente y de dicha velocidad (yv) a la coordenada y, obteniendo así la nueva posición de la pelota.

COMPROBACION DEL COLOR

En las cuatro líneas siguientes se definen cuatro variables, C1 a C4, que corresponden a los números de color de los cuatro cuadrados que rodean a la pelota. A continuación hay dos líneas de programa en las que se hace la comprobación de color; el color 4 es el del contorno y los cuadrados aleatorios.

Si aparece un 4 correspondiendo a alguno de los cuatro cuadrados que rodean a la pelota, la siguiente parte del programa hará que ésta invierta su trayectoria con una dirección opuesta a la que llevaba antes de chocar con el obstáculo. Se hacen dos comprobaciones separadas, una en la dirección horizontal (eje x) y otra en la dirección vertical (eje y). Observa que no se trata simplemente de invertir el movimiento de la pelota en las dos direcciones. Si así se hiciera la pelota siempre se iría por donde había venido.

Las dos comprobaciones para la x, correspondientes a los cuadrados situados a derecha e izquierda de la pelota, se hacen en la misma sentencia IF ... THEN. Se puede hacer esto debido a que el resultado es el mismo si es cierta una cualquiera de las dos condiciones. La componente x de la velocidad se invierte mediante un cambio de signo, pasando de positiva a negativa o viceversa; así −5 se convierte en 5 y 5 se convierte en −5. Lo mismo puede decirse de las comprobaciones para la componente y.

La última línea del programa pokea el valor 160 en la posición antigua de la pelota (contenida en las variables ox y oy), para borrarla de la pantalla. Seguidamente el programa se vuelve a la línea 80 para seguir con el movimiento de la pelota.

Sería muy sencillo añadir unas cuantas líneas al programa anterior para convertirlo en un juego. En futuros artículos de INPUT utilizaremos este comando en una amplia variedad de rutinas de juegos.

Si se te hace difícil encontrar INPUT en tu kiosco habitual, resérvalo por adelantado, o háznoslo saber para que podamos remediarlo



¡¡Arrastre una rodilla por el asfalto!! Aguante el arranque de la carrera a 180 km/h., sin que le rechinen los dientes.

Hay dos pilotos junto a usted. Uno viene por detrás y el otro a la altura de su codo. El rugido de las motos es ensordecedor.

El viento golpea su casco; su adrenalina está subiendo como la espuma.

Tumba su moto hacia la derecha... y ha sido golpeado, está volando y dando tumbos junto con la moto.

Aparte sus ojos de la carretera una milésima de segundo y acabará en la cuneta.

Todo está borroso; no hay tiempo para pensar. La próxima curva acaba en un precipicio y es muy cerrada; los

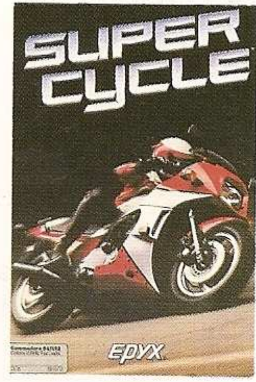


neumáticos de la moto que le precede, hacen saltar grava sobre su... su...

Se oye un teléfono ¿un teléfono? ¡¡Hey!! espere un momento. Esto no es una moto, es una silla. Ahora todo vuelve a la realidad.

Sí, usted está en casa, la pizza está aquí, el ordenador está conectado. Parece, que al final va a ser otra apacible noche después de todo.

Si Super-Cycle fuera un poco más realista, usted necesitaría un seguro para conducirlo.



C-64/128





Santa Cruz de Marcenado, 31 - 28015 Madrid
Telf. 241 10 63

Las pantallas corresponden a la versión C/64/128.

DISTRIBUIDO en Cataluña y Baleares por: DISCLUB, S. A. - Balmes, 58 - BARCELONA - Tel. (93) 302 39 08

ORDENADORES, PUZZLES Y MATEMATICAS

- TIPOS DE PUZZLES
- USO DEL ORDENADOR
- ECUACIONES SIMULTANEAS
- PRUEBA Y ERROR
- TRUCOS MATEMATICOS

Agudiza el ingenio y pon a prueba tu destreza en programación con estos difíciles rompecabezas. También aprenderás aquí unas cuantas técnicas nuevas para resolver sistemas de ecuaciones.

A estas alturas de nuestra andadura en INPUT, ya tienes una buena base en casi todas las técnicas de programación en BASIC. Dado un problema cualquiera, deberías ser capaz de escribir un programa para resolverlo (en el supuesto de que realmente pueda ser resuelto). Pero a menos que tengas un *hobby* especial o algún proyecto entre manos que haga uso del ordenador, con frecuencia es difícil pensar cosas con las que ponerse a hacer pruebas.

Los rompecabezas ofrecen un desafío que suele ser bienvenido, ya que constituyen una forma entretenida de practicar la programación y de mantener al día tus habilidades. Han sido muy populares durante millares de años. Se sabe que los egipcios disfrutaban resolviendo rompecabezas y que los antiguos griegos eran extremadamente aficionados a los rompecabezas y paradojas matemáticas y lógicas. De hecho muchos problemas que empezaron como un simple divertimento condujeron posteriormente a importantes descubrimientos en matemáticas y en otras ciencias.

La resolución de un problema con éxito te hará sentir una sensación de satisfacción y, si participas en alguno de los cada vez más abundantes concursos organizados por las revistas de informática, puedes ganar muchos premios. Pero la solución de estos rompecabezas te enseñará además sobre programación mucho más de lo que puedes aprender por simples lecturas sobre el tema ya que te obliga a trabajar con tus propios métodos y a poner a prueba tus propias ideas.

PUZZLES PARA ORDENADOR

Existen muchos tipos de rompecabezas, no todos de los cuales son adecuados para ser resueltos por medio de ordenador. Muchos de ellos requieren una idea feliz o un poco de razonamiento lógico. Un ejemplo clásico del tipo que requiere razonamiento lógico es el siguiente. Un hombre posee tres cosas: un lobo, una oveja y una berza, las cuales tiene que transportar a la orilla opuesta de un río. Pero el bote que utiliza es muy pequeño y sólo puede llevar una cosa cada vez. Y si deja en tierra el lobo y la oveja, el lobo matará la oveja, mientras que si deja a la oveja y la berza, la oveja se comerá la berza. ¿Cómo puede hacer para pasar a la otra orilla del río las tres cosas y tenerlas intactas?

Este enigma puede resolverse con un ordenador (si te gusta, puedes intentar escribir un programa que lo haga), pero es mucho más rápido hacerlo con papel y lápiz o directamente de memoria.

Los tipos de enigmas que mejor se resuelven por ordenador son los que contienen un cierto número de hechos y cifras, en los que se pide encontrar el valor de alguna incógnita. Otros que también funcionan bien sobre ordenador son los que exigen realizar tareas complicadas de tipo aritmético o geométrico. En tales casos es más rápido escribir un programa que resuelva el problema que intentar hacerlo por tí mismo.

Este artículo te enseñará la forma de resolver tres de los tipos más comunes de *puzzles*. Antes de mirar las soluciones, merece la pena que te entretengas en gastar algo de tiempo intentando resolverlos tú solo. Después pasa ya a consultar los programas y la explicación de su funcionamiento.

SIMPLIFICACION DE PROBLEMAS

Si no estás muy acostumbrado a resolver adivinanzas, al principio puede que la cosa te resulte bastante difícil. Se requiere un poco de práctica para separar la información esencial de la masa de palabras confusas, que sólo sirve para despistar. De hecho los enigmas del primer tipo se basan en las palabras que enmascaran lo que con frecuencia es un problema bastante sencillo. Cuando se le despoja de sus palabras, el enigma conduce habitualmente a la solución de un sistema de ecuaciones. Aquí tienes un sencillo problema que puedes resolver sin ordenador y que te ayudará a entender los principios que intervienen.

Un grupo de amigos entra en un bar y pide tres cafés y dos tés. La cuenta asciende a 176 pesetas. Al día siguiente se les agrega alguien más, y piden el doble de tés y un café menos. Esta vez la cuenta sube 16 pesetas más. ¿Cuánto cuesta la taza de té?

Si eliminas toda la información no esencial y utilizas letras en vez de palabras, el problema se reduce a $3c + 2t = 176$ y $2c + 4t = 192$.

A esto se le llama sistema de ecuaciones simultáneas, porque han de satisfacerse las dos al mismo tiempo. Para poder resolver un sistema de ecuaciones simultáneas se necesitan tantas ecuaciones como incógnitas. En este caso hay dos incógnitas a las que hemos llamado c y t, así como dos ecuaciones, por lo que el sistema en principio tiene solución. Además las ecuaciones son lineales ya que ninguna de las variables aparece elevada a una potencia mayor que 1. Por ejemplo $-3c^2 + 2t = 176$ es una ecuación no lineal que se resuelve por un método diferente.

Para resolver el problema puedes reordenar la segunda ecuación, lo que

te dará c = (192 − 4t)/2 y a continuación sustituir en la primera. Se obtiene 3(192 − 4t)/2 + 2t = 176. Reagrupando de nuevo los términos, se obtiene t = 28, por lo que una taza de té cuesta 28 pesetas.

Las ecuaciones con dos incógnitas como ésta son fáciles de resolver y las de tres incógnitas tampoco son demasiado difíciles. Pero para un mayor número de incógnitas el ordenador resulta decisivo a la hora de hacer por tí el trabajo más duro.

EL VENDEDOR DE SELLOS

Aquí tienes el primer enigma real que debes resolver.

Un vendedor de sellos tiene una caja con sellos extranjeros que piensa vender en paquetes de seis valores diferentes. El precio de los paquetes está codificado desde la A a la F. Seis jóvenes miembros de un club filatèlico se presentan a la vez dispuestos a gastarse todo el dinero que llevan en el bolsillo como se ve en el cuadro 1.

	A	B	C	D	E	F	TOTAL
Elena	6	2	3	1	1	2	341
Jaime	14	11	0	1	2	1	469
Anita	0	1	3	6	4	3	598
Mar	5	3	5	2	1	1	376
Luis	12	1	4	4	3	2	587
Clara	8	0	1	1	0	3	293

La pregunta es: ¿Cuál era el precio de cada paquete?

El intentar resolver este sistema de ecuaciones eliminando una variable en cada paso, es posible, pero requeriría mucho tiempo y se presta a que se cometan errores de cálculo. Con el siguiente programa podrás resolver esta adivinanza así como cualquier otro sistema de ecuaciones lineales simultáneas, las cuales se presentan en muchos tipos de problemas y no sólo en la resolución de acertijos.

Existen varias formas de resolver sistemas de ecuaciones como éste, y puedes encontrar métodos diferentes, pero el programa que sigue es rápido y relativamente corto:

Teclea para C-64 y Vic-20

```
10 PRINT"[SHIFT+CLR/HOME]":
   INPUT"INTRODUCE
   EL NO. DE FILAS";R:C=R+1:
   PRINT
20 DIM A(R,C),B
   (R,C),A$(C-1)
30 FOR K=1 TO C-1
40 PRINT"NOMBRE DE LA
   INCOGNITA";K;:INPUT
   A$(K)
50 NEXT:PRINT
60 FOR J=1 TO R
70 PRINT"COEFICIENTES DE LA
   FILA";J
```



```
80 FOR K=1 TO C
90 INPUT A(J,K)
100 B(J,K)=A(J,K)
110 NEXT:NEXT
120 FOR L=1 TO R
130 GOSUB 230
140 GOSUB 280
150 NEXT
160 PRINT"[SHIFT+CLR/HOME]"
170 FOR K=1 TO C-1:PRINT TAB
    (8*K-7);A$(K);:NEXT:PRINT
180 FOR J=1 TO R:FOR K=1TOC:
    PRINT TAB(K*8-8);B(J,K);
190 NEXT:PRINT:NEXT
200 PRINT:PRINT"RESULTADOS:-"
210 FOR K=1 TO C-1:PRINT TAB
    (8*K-8);A(K,C);:NEXT
220 END
230 REM
240 D=A(L,L)
250 FOR K=1 TO C
260 A(L,K)=A(L,K)/D
270 NEXT:RETURN
280 REM
290 FOR J=1 TO R
300 IF J=L THEN NEXT:RETURN
310 F=A(J,L)
320 FOR K=1 TO C
330 A(J,K)=A(J,K)-F*A(L,K)
340 NEXT:NEXT:RETURN
```

La primera parte del programa, entre las líneas 10 y 110, te permite introducir los datos, mientras que la segunda parte, situada entre las líneas 120 y 150 llama a dos subrutinas (o procedimientos) para calcular la respuesta, la cual se imprime entonces en las líneas 160 a 220.

Los valores se cargan en la matriz A(J,K) a razón de una fila cada vez. En este programa, J se refiere a las filas y K a las columnas, lo cual has de tener siempre en la mente al leer el resto del programa. La matriz A(J,K) se copia sobre una matriz idéntica B(J,K) de forma que se pueden imprimir los valores originales junto con la respuesta en la línea 180.

El cálculo avanza de fila en fila, controlado por el bucle que se extiende entre las líneas 120 y 150. Antes de hacer otras cosas, la rutina de las líneas 230 a 270 extrae el elemento diagonal de la fila (es decir A(1,1), A(2,2), etc.) y divide cada elemento de dicha fila por ese valor. El valor del elemento diagonal resultante es ahora igual a uno, es decir, ha quedado normalizado.

La siguiente rutina es la encargada de hacer la mayor parte del trabajo de este programa. Aunque sólo contiene seis líneas, resulta muy difícil examinar lo que hace. La mejor manera de entenderlo es examinar a fondo un ejemplo real (aunque corto) siguiendo uno por uno los pasos que va dando el ordenador. En esencia funciona de la siguiente manera. Supongamos que ya has normalizado la fila 1, por lo que L es 1. La línea 290 toma entonces cada fila diferente de la 1 (línea 300), por lo que en este caso empieza por la 2. La línea 310 toma el primer elemento de la fila 2 y le llama F. Todavía en la fila 2, la línea 320 va recorriendo todas sus columnas, y la línea 330 multiplica F por el elemento correspondiente de cada columna, pero de la fila 1, restándolo del elemento de la misma columna en la fila 2. Esto mismo lo va haciendo con las filas 3, 4, 5 y 6. Seguidamente empieza otra vez con L igual a 2.

Al final de este proceso, los elementos diagonales de cada fila siguen siendo y todos los demás elementos son ceros. Por eso se pueden leer con facilidad los valores en la columna de la derecha.

LOS REGALOS DE NAVIDAD

El programa que acabamos de ver sirve para resolver cualquier sistema de ecuaciones lineales simultáneas en el que hay tantas ecuaciones como inen los que las cosas se organizan de tal manera que no hay ecuaciones suficientes, por lo que no se pueden resolver de esta forma. De hecho no existe una solución única y el truco consiste en seleccionar la respuesta más plausible entre todas las soluciones disponibles. Normalmente el enigma contendrá una clave que te ayudará. También podría ocurrir que las ecuaciones resultaran ser no lineales.

Ensaya ahora el siguiente enigma. En Navidad, el tío Alberto, que es un matemático un poco excéntrico, explicó a sus dos jóvenes sobrinos que le daría a cada uno tantos paquetes como la edad que tenían (contando sólo los años). También les dijo que cada paquete contenía el mismo número de sobres que la edad de cada uno, y que cada sobre contenía un número de pesetas igual a la edad de cada niño. Después de preparar los regalos, se le oyó murmurar entre dientes que el siguiente año la cosa le iba a costar 500 pesetas más. ¿Qué edades tenían los dos niños?

Suprimiendo las palabras y llamando A y B a las edades de los niños, así como M a la cantidad de dinero gastada en este año, la solución del enigma queda reducida a la de las ecuaciones siguientes:

A ↑ 3 + B ↑ 3 = M;
(A + 1) ↑ 3 + (B + 1) ↑ 3 = M + 500

Tenemos tres incógnitas: A, B y M, pero sólo dos ecuaciones. Sin tener un ordenador, podrías utilizar un método de prueba y error para ir ensayando diferentes valores de A y B, y observando si las ecuaciones se satisfacen. El ordenador trabaja de una forma parecida, salvo que naturalmente puede hacer varios cientos o miles de ensayos de valores diferentes en un tiempo muy corto y sin cometer errores.

El primer paso en la resolución del problema es buscar en el enunciado del enigma alguna pista suplementaria que te pueda ayudar a limitar el margen de valores que debes ensayar. Como se habla de «jóvenes» sobrinos, no es probable que tengan más de 14 años, y es muy probable que al menos tengan tres años. El programa para resolver el problema es muy corto:

Teclea para C-64 y Vic-20

```
10 FOR A=3 TO 14
20 FOR B=A TO 14
30 M=A^3+B^3
40 N=(A+1)^3+(B+1)^3
50 IF ABS(M+500-N)<.01 THEN
   PRINT"A=";A,"B=";B
60 NEXT B
70 NEXT A
```

ECUACIONES SINGULARES

Hay algunas situaciones especiales, con frecuencia bastante tontas, en que los sistemas de ecuaciones simultáneas tienen o una solución ambigua o carecen por completo de solución. Las ecuaciones x = 2 y x = 3 obviamente no pueden satisfacerse al mismo tiempo, e incluso cuando tiene tantas ecuaciones como incógnitas, como en el caso x + y = 3; x + y = 1 no pueden cumplirse simultáneamente.

Otro caso son las ecuaciones x + y = 2; 2x + 2y = 4. En este caso se trata de dos ecuaciones que no son independientes; esencialmente se trata de la misma ecuación, existiendo muchas soluciones posibles.

Cuando cuentes con el número de ecuaciones adecuado y siga habiendo incompatibilidades o las ecuaciones no sean independientes, se dice que el sistema es singular.

El caso de las ecuaciones dependientes se puede resolver utilizando un método de prueba y error, como hacíamos en el caso de los regalos de Navidad. Pero al ser posibles varias soluciones necesitas alguna información adicional para seleccionar la correcta.

Al ejecutar este programa obtendrás solamente una respuesta, debido a las restricciones introducidas en las líneas 10 y 20. Si en el enigma no se hubiera especificado que los sobrinos son jóvenes, el bucle FOR ... NEXT tendría que extenderse más. Digamos de paso que la condición impuesta en la línea 50 evita problemas con los errores de redondeo; lo que realmente se está comprobando es si se cumple que M + 50 es igual a N.

Este método debe funcionar para todos los problemas con más incógnitas que ecuaciones. En otros tipos de problemas, puede que te sea necesario hacer hipótesis acerca de los valores de más variables, o tal vez incluir más condiciones IF ... THEN del tipo de la que aparece en la línea 50. Con un número muy elevado de incógnitas o de condiciones el programa puede tardar muchos minutos o incluso horas en ejecutarse, aunque seguirá siendo resoluble.

MAGIA MATEMATICA

El tercer tipo de rompecabezas que aparecen en las revistas de informática son los problemas de números. Suelen plantear cuestiones de aritmética aparentemente sencillas que resultan extraordinariamente laboriosas de resolver a menos que se utilice un ordenador.

Aquí tienes un par de ejemplos.

Existe un número de cuatro cifras que cuando se invierte y se multiplica por un entero se convierte en el número original: 8712 = 4*2178. El problema es encontrar si existen más números con esta misma propiedad, y en caso afirmativo calcularlos.

Otro número interesante es el 987654321. Este número es un múltiplo exacto de 17 y utiliza todos los dígitos del 1 al 9. Trata de encontrar el número inferior más proximo que tenga estas mismas propiedades.

Las soluciones a este tipo de problemas se basan en considerar el número como una colección de cifras más que como un valor numérico. Puedes, dividir el número en unidades, decenas, centenas, etc, o tratarlo como una cadena de cifras y separar éstas con ayuda de los comandos de tratamiento de cadenas.

El primer problema se resuelve mediante el primer método. Si representamos el número por ABCD, da origen a la siguiente ecuación:

1000*A + 100*B + 10*C + D = X*(1000*D + 100*C + 10*B + A)

cuyas incógnitas tienes que calcular. Como de costumbre la principal dificultad consiste en decidir qué valores se ensayan. A puede variar entre 1 y 9, no pudiendo ser cero ya que en ese caso tendrías un número de tres cifras. B y C pueden variar entre 0 y 9. El factor de multiplicación X debe ser al menos 2 y no debe superar a 10/D, para que el segundo miembro de la ecuación sea un número de cinco cifras. Por la misma razón D no puede ser mayor que 4. Ahora puedes ya escribir el programa para resolver el problema:

```
10 FOR A=1 TO 9
20 FOR B=0 TO 9
30 FOR C=0 TO 9
40 FOR D=1 TO 4
50 FOR X=2 TO INT(9.9/D)
60 J=1000*A+100*B+10*C+D
70 K=1000*D+100*C+10*B+A
80 IF X*K=J THEN PRINT J;
   "=";X;"*";K
90 NEXT X,D,C,B,A
```

Teclea el programa, a continuación teclea RUN y siéntate a esperar porque llevará bastante tiempo la comprobación de todas las combinaciones posibles.

La solución del segundo problema trata a los números como cadenas.

Programa 4 S ó 4 C

```
10 M=987654321
20 M=M-17
30 M$=STR$(M)
40 F=0
50 FOR P=1 TO 9
60 P$=CHR$(48+P)
65 FOR Z=1 TO LEN(M$)
70 IF MID$(M$,Z,1)=P$ THEN
   90
80 NEXT Z:F=F+1
90 NEXT P:IF F=0 THEN PRINT
   M$:END
100 GOTO 20
```

El programa parte de 987654321 y va contando hacia abajo de 17 en 17, convirtiendo cada múltiplo de 17 en una cadena de números a la que llama M$. Las líneas 50 y 60 convierten cada dígito del 1 al 9 en una cadena y a continuación la línea 70 comprueba si dicha cadena figura en M$. Si hay algún dígito que no esté en M$, hay un indicador F que se incrementa una unidad. Sólo se imprime M$ cuando contiene todos los dígitos del 1 al 9.

CURSO DE

BASIC + MICROORDENADORES

prácticas con...

Microordenador ZX SPECTRUM

Microordenador COMMODORE





Para saber cómo hablar con los ordenadores

El Curso CEAC a Distancia, BASIC + Microordenadores, le va a introducir paso a paso, con un cuidado método, en uno de los temas más apasionantes de nuestros días: **la programación de ordenadores.** Al aprender PRACTICANDO desde un principio a programar BASIC, lenguaje diseñado especialmente para dar los primeros pasos en programación, estará sentando las bases para el estudio de cualquier otro lenguaje de alto nivel.

Curso CEAC de BASIC + Microordenadores: un diálogo permanente con el ordenador.

Otros Cursos:

- Introducción a la Informática
- Electrónica (con experimento
- Contabilidad
- Fotografía
- Curso de Video
- Decoración

ESTAS ENSEÑANZAS SE AJUSTAN AL ART. 35 DEL DECRETO 707/1976 Y A LA ORDEN MINISTERIAL DE 5/2/1979

Actúe ahora en su propio beneficio y pídanos información.

GRATUITAMENTE

Sí, deseo recibir a la mayor brevedad posible información sobre el Curso de: ______

Nombre y apellidos ______ Edad ______

Domicilio ______

______ Nº ______ Piso ______ Pta. ______ Tel. ______

C. Postal ______ Población ______

______ Provincia ______

Profesión ______

CEAC. Aragón, 472
(Dpto. T-ZF) 08013 Barcelona



CEAC

CENTRO DE ENSEÑANZA A DISTANCIA AUTORIZADO POR EL MINISTERIO DE EDUCACIÓN Y CIENCIA N.º 8039185
(BOLETÍN OFICIAL DEL ESTADO 3-6-83)
Aragón, 472 (Dpto. T-ZF) 08013 Barcelona
Tel.: (93) 245 33 06

COMO HACER REFERENCIAS CRUZADAS

- SEGUIMIENTO DE ERRORES DE PROGRAMACION
- USO DEL PROGRAMA
- BUSQUEDA DE CADENAS DE CARACTERES
- BUSQUEDA CON SUSTITUCION
- LISTADO DE FUNCIONES Y PROCEDIMIENTOS



Con esta práctica utilidad en código máquina podrás localizar en pocos segundos cualquier cadena de caracteres o cualquier palabra reservada dentro de un programa en BASIC, ahorrándote gran cantidad de tiempo y esfuerzo. También podrás hacer la sustitución de una cadena por otra.

El programa de referencias cruzadas que presentamos en este artículo es lo que se llama un programa de utilidades o simplemente una «utilidad». Esto significa que no hace nada él solo, sino que se utiliza como una herramienta para ayudarte a desarrollar otros programas. Por esta razón está escrito en lenguaje máquina, a fin de que pueda ser cargado en el ordenador al mismo tiempo que un programa en BASIC. El programa en lenguaje máquina se queda en la memoria del ordenador esperando a que se le llame cada vez que haga falta su ayuda.

El programa se utiliza para realizar búsquedas de textos, con lo que te podrás librar de tener que seguir penosamente la pista a alguna determinada sección de un programa. Si le dices lo que estás buscando —una cadena de caracteres, una palabra reservada, un nombre de variable o lo que sea— el programa se encargará de buscarlo a lo largo del programa en BASIC y presentará todas las líneas en las que aparezca. Aparte de esto, también cuenta con una facilidad extra de «sustitución» que te permitirá remplazar una cadena por otra a lo largo de todo el programa.

El programa te resultará de la mayor utilidad cuando estés desarrollando o escribiendo un programa en BASIC. Por ejemplo, si llegas a la conclusión de que tienes que cambiar algunas de las sentencias PRINT o INPUT, el programa te listará todas las líneas en que aparezcan dichas sentencias. O si te das cuenta de que el nombre de una de las variables de tu programa entra en conflicto con otras rutinas, puedes usar la opción de búsqueda con sustitución, cambiando el nombre por otro, sin tener que editar ni una línea.

También te resultará muy útil tener cargado en el ordenador el programa de las referencias cruzadas cuando estés intentando adaptar o comprender un programa escrito por otra persona. Por ejemplo, puede ser un ejercicio muy fructífero imprimir todas las líneas en que aparecen las palabras GOTO o GOSUB, ya que a partir de aquí rápidamente podrás trazar la secuencia de sucesos.

También te ayudará a localizar y perseguir los errores de programación que pueda contener tu programa. Por ejemplo, puede ser que descubras que una determinada variable tome valores incorrectos en alguna parte de tu programa. El procedimiento usual de trabajo en estos casos es buscar línea por línea, a lo largo de todo el programa para localizar todos los sitios en que aparece dicha variable; es más que probable que se te escapen unas cuantas. Sin embargo el programa de referencias cruzadas hará este trabajo por tí con toda precisión en cuestión de segundos.

El programa de referencias cruzadas forma parte de un paquete más amplio de herramientas de programación. El programa del que nos ocupamos aquí contiene un corto programa cargador escrito en BASIC y las sentencias DATA que contienen el código máquina. Teclea el programa y almacénalo antes de ejecutarlo, por si se te rompe al hacerlo. Seguidamente teclea RUN y, suponiendo que todo vaya bien, almacena el código máquina resultante tecleando:

Teclea para Commodore-64

```
POKE 43,0:POKE 44,192:POKE
```




```
45,47:POKE 46,195:CLR
SAVE" REFCRUZ",1,1
```

Cuando llegue el momento de utilizar el programa necesitas cargar esta versión del código máquina. Para hacerlo, pulsa:

```
LOAD "REFCRUZ",1,1
```

Tienes que hacer esta carga al principio de una sesión de programación, antes de cargar un programa en BASIC. En cualquier momento puedes llamar al programa en código máquina utilizando SYS 49152, con lo que te aparecerán las opciones disponibles. Recuerda que si estás utilizando una unidad de discos tienes que cambiar por un 8 el primer 1 que figura en los comandos LOAD y SAVE anteriores.

Para buscar una determinada cadena de caracteres, teclea L (de List) seguida de los caracteres que quieres buscar, hasta un máximo de 76, y a continuación pulsa RETURN . El programa te presentará todas las líneas en que aparece dicha cadena de caracteres.

Para reemplazar una cadena por otra, pulsa R seguida de la cadena original, a continuación una barra inclinada (/) y la cadena que habrá de sustituirla, por ejemplo R ab/zz. Se pueden sustituir hasta cuatro caracteres y la nueva cadena puede ser más corta, pero no más larga que la original.

Si tecleas K seguido del nombre de una palabra reservada, te aparecerán listadas todas las líneas en que contengan dicha palabra.

Si tecleas F seguida del nombre de una función definida por el usuario, el programa listará todas las líneas en que dicha función es definida o llamada.

Por último, para salir del programa y regresar al BASIC, pulsa E.

Teclea para Commodore-64

```
0 DATA 234,169,62,32,12,225,
  169,0,133,247,133,79,133,
  100,169,#1897
10 DATA 8,133,101,32,18,225,
  201,76,240,50,201,82,240,
  107,201,#1915
20 DATA 70,240,31,201,69,240
  ,26,201,75,240,99,169,69,
  32,12,#1774
30 DATA 225,169,82,32,12,225
  ,32,12,225,169,13,32,12,
  225,76,#1541
40 DATA 0,192,234,96,234,162
  ,0,169,165,149,85,232,76,
  78,192,#2064
50 DATA 234,162,0,234,32,38,
  194,160,0,234,177,100,201
  ,0,208,#1974
60 DATA 7,32,158,192,201,0,
  240,158,234,162,0,213,85,
  208,24,#1914
70 DATA 165,101,133,102,132,
  103,234,228,96,240,86,232
  ,32,28,194,#2106
80 DATA 177,100,213,85,208,
  3,76,111,192,234,32,28,
  194,76,84,#1813
90 DATA 192,234,76,60,194,
  162,0,32,18,225,201,13,
  240,6,149,#1802
100 DATA 85,232,76,142,192,
  76,202,194,234,32,28,194
  ,177,100,201,#2165
110 DATA 0,208,28,32,28,194,
  177,100,201,0,240,18,32,
  28,194,#1480
120 DATA 165,101,133,95,152,
  133,94,32,28,194,32,28,
  194,177,100,#1658
130 DATA 96,32,28,194,76,177
  ,192,234,165,247,201,1,
  208,3,32,#1886
140 DATA 127,194,165,94,168,
  165,95,133,101,169,0,133
  ,100,177,100,#1921
150 DATA 133,252,32,28,194,
  177,100,133,253,72,138,
  72,152,72,32,#1840
160 DATA 181,193,104,168,
  104,170,104,32,28,194,
  234,177,100,201,34,#2024
170 DATA 240,38,201,0,240,25
  ,201,255,240,18,234,201,
  127,176,49,#2245
180 DATA 201,0,176,9,32,28,
  194,76,250,192,76,0,192,
  76,171,#1673
190 DATA 193,234,169,13,32,
  12,225,76,84,192,32,12,
  225,165,79,#1743
200 DATA 201,1,240,5,169,1,
  76,55,193,169,0,133,79,
  32,28,#1382
210 DATA 194,76,250,192,234,
  165,79,201,0,240,11,177,
  100,32,12,#1963
220 DATA 225,32,28,194,76,
  250,192,177,100,56,233,
  128,170,152,72,#2085
230 DATA 160,0,169,160,133,
  255,169,158,133,254,234,
  224,0,240,30,#2319
240 DATA 177,254,201,128,176
  ,11,192,255,208,2,230,
  255,234,200,76,#2599
250 DATA 99,193,234,202,192,
  255,208,2,230,255,234,
  200,76,99,193,#2672
260 DATA 234,24,177,254,201,
  128,176,14,32,12,225,192
  ,255,208,2,#2134
270 DATA 230,255,234,200,76,
  134,193,234,24,233,127,
  32,12,225,104,#2313
280 DATA 168,32,28,194,76,
  250,192,234,32,12,225,32
  ,28,194,76,#1773
290 DATA 250,192,234,160,0,
  234,162,0,169,0,133,78,
  234,165,253,#2264
300 DATA 201,0,208,6,165,252
  ,201,10,144,30,234,165,
  252,56,233,#2157
310 DATA 10,133,252,165,253,
  233,0,133,253,224,255,
  240,4,232,76,#2463
320 DATA 191,193,234,230,78,
  162,0,76,191,193,234,165
  ,252,105,48,#2352
330 DATA 72,165,78,133,253,
  201,0,208,11,138,200,201
  ,10,144,13,#1827
340 DATA 133,252,76,184,193,
  234,138,200,133,252,76,
  184,193,234,105,#2587
350 DATA 48,32,12,225,234,
  104,32,12,225,136,192,0,
  208,246,96,#1802
360 DATA 234,192,255,208,2,
  230,101,234,200,96,234,
  32,18,225,201,#2462
370 DATA 13,240,6,149,85,232
  ,76,38,194,234,202,134,
  96,96,76,#1871
380 DATA 0,192,234,169,1,133
  ,247,162,0,234,32,18,225
  ,201,47,#1895
390 DATA 240,6,149,85,232,76
  ,67,194,234,202,134,96,
  162,0,234,#2111
400 DATA 32,18,225,201,13,
  240,6,149,89,232,76,87,
  194,234,202,#1998
410 DATA 228,96,240,12,176,
  10,169,32,234,232,149,
  89,228,96,208,#2199
420 DATA 248,234,134,83,160,
  0,76,84,192,234,24,165,
  102,133,101,#1970
430 DATA 164,103,162,0,234,
  181,89,145,100,228,96,
  240,10,232,32,#2016
440 DATA 28,194,76,137,194,
  76,57,194,234,162,0,32,
  28,194,177,#1783
450 DATA 100,213,85,240,6,
```

```
    201,0,208,240,234,96,234
    ,165,101,133,#2256
460 DATA 102,132,103,234,232
    ,32,28,194,177,100,213,
    85,208,220,228,#2288
470 DATA 96,240,187,76,182,
    194,76,57,194,169,0,133,
    251,169,160,#2184
480 DATA 133,255,169,158,133
    ,254,160,0,162,0,177,254
    ,213,85,240,#2393
490 DATA 35,201,128,176,49,
    192,255,208,2,230,255,
    200,177,254,201,#2563
500 DATA 128,144,243,192,
    255,208,2,230,255,200,
    230,251,165,251,201,
    #2955
510 DATA 255,240,199,76,216,
    194,192,255,208,2,230,
    255,200,232,177,#2931
520 DATA 254,213,85,240,242,
    201,128,144,207,233,128,
    213,85,208,214,#2795
530 DATA 165,251,24,105,128,
    133,251,162,0,149,85,169
    ,0,76,82,#1780
540 DATA 192
600 PRINT"[SHIFT+CLR/HOME]":
    M=49152:FOR Z=0 TO 53:
    C=0:FOR ZZ=0 TO 14:READ
    X:POKE M,X
610 PRINT"[CRSR arriba]LINEA
    DATA ";PEEK(63)+256*PEEK
    (64):C=C+X:M=M+1:NEXT ZZ
    :READ X$
620 IF VAL(RIGHT$(X$,LEN(X$)
    -1))<>C THEN PRINT"ERROR
    : VERIFICAR LA LINEA!":
    END
630 NEXT Z:READ X:POKE M,X:
    PRINT"[SHIFT+CLR/HOME]
    DATAS OK"
```

Si prefieres utilizar un ensamblador o el cartucho HESMON, utiliza el siguiente listado equivalente

```
 DIR.  HEXADEC.  NEMONICO
=====  =======   ========

,C000 EA          NOP
,C001 A9 3E       LDA #$3E
,C003 20 0C E1    JSR $E10C
,C006 A9 00       LDA #$00
,C008 85 F7       STA $F7
,C00A 85 4F       STA $4F
,C00C 85 64       STA $64
,C00E A9 08       LDA #$08
,C010 85 65       STA $65
,C012 20 12 E1    JSR $E112
,C015 C9 4C       CMP #$4C
,C017 F0 32       BEQ $C04B
,C019 C9 52       CMP #$52
,C01B F0 6B       BEQ $C088
,C01D C9 46       CMP #$46
,C01F F0 1F       BEQ $C040
,C021 C9 45       CMP #$45
,C023 F0 1A       BEQ $C03F
,C025 C9 4B       CMP #$4B
,C027 F0 63       BEQ $C08C
,C029 A9 45       LDA #$45
,C02B 20 0C E1    JSR $E10C
,C02E A9 52       LDA #$52
,C030 20 0C E1    JSR $E10C
,C033 20 0C E1    JSR $E10C
,C036 A9 0D       LDA #$0D
,C038 20 0C E1    JSR $E10C
,C03B 4C 00 C0    JMP $C000
,C03E EA          NOP
,C03F 60          RTS
,C040 EA          NOP
,C041 A2 00       LDX #$00
,C043 A9 A5       LDA #$A5
,C045 95 55       STA $55,X
,C047 E8          INX
,C048 4C 4E C0    JMP $C04E
,C04B EA          NOP
,C04C A2 00       LDX #$00
,C04E EA          NOP
,C04F 20 26 C2    JSR $C226
,C052 A0 00       LDY #$00
,C054 EA          NOP
,C055 B1 64       LDA ($64),Y
,C057 C9 00       CMP #$00
,C059 D0 07       BNE $C062
,C05B 20 9E C0    JSR $C09E
,C05E C9 00       CMP #$00
,C060 F0 9E       BEQ $C000
,C062 EA          NOP
,C063 A2 00       LDX #$00
,C065 D5 55       CMP $55,X
,C067 D0 18       BNE $C081
,C069 A5 65       LDA $65
,C06B 85 66       STA $66
,C06D 84 67       STY $67
,C06F EA          NOP
,C070 E4 60       CPX $60
,C072 F0 56       BEQ $C0CA
,C074 E8          INX
,C075 20 1C C2    JSR $C21C
,C078 B1 64       LDA ($64),Y
,C07A D5 55       CMP $55,X
,C07C D0 03       BNE $C081
,C07E 4C 6F C0    JMP $C06F
,C081 EA          NOP
,C082 20 1C C2    JSR $C21C
,C085 4C 54 C0    JMP $C054
,C088 EA          NOP
,C089 4C 3C C2    JMP $C23C
,C08C A2 00       LDX #$00
,C08E 20 12 E1    JSR $E112
,C091 C9 0D       CMP #$0D
,C093 F0 06       BEQ $C09B
,C095 95 55       STA $55,X
,C097 E8          INX
,C098 4C 8E C0    JMP $C08E
,C09B 4C CA C2    JMP $C2CA
,C09E EA          NOP
,C09F 20 1C C2    JSR $C21C
,C0A2 B1 64       LDA ($64),Y
,C0A4 C9 00       CMP #$00
,C0A6 D0 1C       BNE $C0C4
,C0A8 20 1C C2    JSR $C21C
,C0AB B1 64       LDA ($64),Y
,C0AD C9 00       CMP #$00
,C0AF F0 12       BEQ $C0C3
,C0B1 20 1C C2    JSR $C21C
,C0B4 A5 65       LDA $65
,C0B6 85 5F       STA $5F
,C0B8 98          TYA
,C0B9 85 5E       STA $5E
,C0BB 20 1C C2    JSR $C21C
,C0BE 20 1C C2    JSR $C21C
,C0C1 B1 64       LDA ($64),Y
,C0C3 60          RTS
,C0C4 20 1C C2    JSR $C21C
,C0C7 4C B1 C0    JMP $C0B1
,C0CA EA          NOP
,C0CB A5 F7       LDA $F7
,C0CD C9 01       CMP #$01
,C0CF D0 03       BNE $C0D4
,C0D1 20 7F C2    JSR $C27F
,C0D4 A5 5E       LDA $5E
,C0D6 A8          TAY
,C0D7 A5 5F       LDA $5F
,C0D9 85 65       STA $65
,C0DB A9 00       LDA #$00
,C0DD 85 64       STA $64
,C0DF B1 64       LDA ($64),Y
,C0E1 85 FC       STA $FC
,C0E3 20 1C C2    JSR $C21C
,C0E6 B1 64       LDA ($64),Y
```

```
,C0E8 85 FD     STA $FD
,C0EA 48        PHA
,C0EB 8A        TXA
,C0EC 48        PHA
,C0ED 98        TYA
,C0EE 48        PHA
,C0EF 20 B5 C1  JSR $C1B5
,C0F2 68        PLA
,C0F3 A8        TAY
,C0F4 68        PLA
,C0F5 AA        TAX
,C0F6 68        PLA
,C0F7 20 1C C2  JSR $C21C
,C0FA EA        NOP
,C0FB B1 64     LDA ($64),Y
,C0FD C9 22     CMP #$22
,C0FF F0 26     BEQ $C127
,C101 C9 00     CMP #$00
,C103 F0 19     BEQ $C11E
,C105 C9 FF     CMP #$FF
,C107 F0 12     BEQ $C11B
,C109 EA        NOP
,C10A C9 7F     CMP #$7F
,C10C B0 31     BCS $C13F
,C10E C9 00     CMP #$00
,C110 B0 09     BCS $C11B
,C112 20 1C C2  JSR $C21C
,C115 4C FA C0  JMP $C0FA
,C118 4C 00 C0  JMP $C000
,C11B 4C AB C1  JMP $C1AB
,C11E EA        NOP
,C11F A9 0D     LDA #$0D
,C121 20 0C E1  JSR $E10C
,C124 4C 54 C0  JMP $C054
,C127 20 0C E1  JSR $E10C
,C12A A5 4F     LDA $4F
,C12C C9 01     CMP #$01
,C12E F0 05     BEQ $C135
,C130 A9 01     LDA #$01
,C132 4C 37 C1  JMP $C137
,C135 A9 00     LDA #$00
,C137 85 4F     STA $4F
,C139 20 1C C2  JSR $C21C
,C13C 4C FA C0  JMP $C0FA
,C13F EA        NOP
,C140 A5 4F     LDA $4F
,C142 C9 00     CMP #$00
,C144 F0 0B     BEQ $C151
,C146 B1 64     LDA ($64),Y
,C148 20 0C E1  JSR $E10C
,C14B 20 1C C2  JSR $C21C
,C14E 4C FA C0  JMP $C0FA
,C151 B1 64     LDA ($64),Y
,C153 38        SEC
,C154 E9 80     SBC #$80
,C156 AA        TAX
,C157 98        TYA
,C158 48        PHA
,C159 A0 00     LDY #$00
,C15B A9 A0     LDA #$A0
,C15D 85 FF     STA $FF
,C15F A9 9E     LDA #$9E
,C161 85 FE     STA $FE
,C163 EA        NOP
,C164 E0 00     CPX #$00
,C166 F0 1E     BEQ $C186
,C168 B1 FE     LDA ($FE),Y
,C16A C9 80     CMP #$80
,C16C B0 0B     BCS $C179
,C16E C0 FF     CPY #$FF
,C170 D0 02     BNE $C174
,C172 E6 FF     INC $FF
,C174 EA        NOP
,C175 C8        INY
,C176 4C 63 C1  JMP $C163
,C179 EA        NOP
,C17A CA        DEX
,C17B C0 FF     CPY #$FF
,C17D D0 02     BNE $C181
,C17F E6 FF     INC $FF
,C181 EA        NOP
,C182 C8        INY
,C183 4C 63 C1  JMP $C163
,C186 EA        NOP
,C187 18        CLC
,C188 B1 FE     LDA ($FE),Y
,C18A C9 80     CMP #$80
,C18C B0 0E     BCS $C19C
,C18E 20 0C E1  JSR $E10C
,C191 C0 FF     CPY #$FF
,C193 D0 02     BNE $C197
,C195 E6 FF     INC $FF
,C197 EA        NOP
,C198 C8        INY
,C199 4C 86 C1  JMP $C186
,C19C EA        NOP
,C19D 18        CLC
,C19E E9 7F     SBC #$7F
,C1A0 20 0C E1  JSR $E10C
,C1A3 68        PLA
,C1A4 A8        TAY
,C1A5 20 1C C2  JSR $C21C
,C1A8 4C FA C0  JMP $C0FA
,C1AB EA        NOP
,C1AC 20 0C E1  JSR $E10C
,C1AF 20 1C C2  JSR $C21C
,C1B2 4C FA C0  JMP $C0FA
,C1B5 EA        NOP
,C1B6 A0 00     LDY #$00
,C1B8 EA        NOP
,C1B9 A2 00     LDX #$00
,C1BB A9 00     LDA #$00
,C1BD 85 4E     STA $4E
,C1BF EA        NOP
,C1C0 A5 FD     LDA $FD
,C1C2 C9 00     CMP #$00
,C1C4 D0 06     BNE $C1CC
,C1C6 A5 FC     LDA $FC
,C1C8 C9 0A     CMP #$0A
,C1CA 90 1E     BCC $C1EA
,C1CC EA        NOP
,C1CD A5 FC     LDA $FC
,C1CF 38        SEC
,C1D0 E9 0A     SBC #$0A
,C1D2 85 FC     STA $FC
,C1D4 A5 FD     LDA $FD
,C1D6 E9 00     SBC #$00
,C1D8 85 FD     STA $FD
,C1DA E0 FF     CPX #$FF
,C1DC F0 04     BEQ $C1E2
,C1DE E8        INX
,C1DF 4C BF C1  JMP $C1BF
,C1E2 EA        NOP
,C1E3 E6 4E     INC $4E
,C1E5 A2 00     LDX #$00
,C1E7 4C BF C1  JMP $C1BF
,C1EA EA        NOP
,C1EB A5 FC     LDA $FC
,C1ED 69 30     ADC #$30
,C1EF 48        PHA
,C1F0 A5 4E     LDA $4E
,C1F2 85 FD     STA $FD
,C1F4 C9 00     CMP #$00
,C1F6 D0 0B     BNE $C203
,C1F8 8A        TXA
,C1F9 C8        INY
,C1FA C9 0A     CMP #$0A
,C1FC 90 0D     BCC $C20B
,C1FE 85 FC     STA $FC
,C200 4C B8 C1  JMP $C1B8
,C203 EA        NOP
,C204 8A        TXA
,C205 C8        INY
,C206 85 FC     STA $FC
,C208 4C B8 C1  JMP $C1E8
,C20B EA        NOP
,C20C 69 30     ADC #$30
,C20E 20 0C E1  JSR $E10C
,C211 EA        NOP
,C212 68        PLA
,C213 20 0C E1  JSR $E10C
,C216 88        DEY
,C217 C0 00     CPY #$00
,C219 D0 F6     BNE $C211
,C21B 60        RTS
```

```
,C21C EA        NOP
,C21D C0 FF     CPY #$FF
,C21F D0 02     BNE $C223
,C221 E6 65     INC $65
,C223 EA        NOP
,C224 C8        INY
,C225 60        RTS
,C226 EA        NOP
,C227 20 12 E1  JSR $E112
,C22A C9 0D     CMP #$0D
,C22C F0 06     BEQ $C234
,C22E 95 55     STA $55,X
,C230 E8        INX
,C231 4C 26 C2  JMP $C226
,C234 EA        NOP
,C235 CA        DEX
,C236 86 60     STX $60
,C238 60        RTS
,C239 4C 00 C0  JMP $C000
,C23C EA        NOP
,C23D A9 01     LDA #$01
,C23F 85 F7     STA $F7
,C241 A2 00     LDX #$00
,C243 EA        NOP
,C244 20 12 E1  JSR $E112
,C247 C9 2F     CMP #$2F
,C249 F0 06     BEQ $C251
,C24B 95 55     STA $55,X
,C24D E8        INX
,C24E 4C 43 C2  JMP $C243
,C251 EA        NOP
,C252 CA        DEX
,C253 86 60     STX $60
,C255 A2 00     LDX #$00
,C257 EA        NOP
,C258 20 12 E1  JSR $E112
,C25B C9 0D     CMP #$0D
,C25D F0 06     BEQ $C265
,C25F 95 59     STA $59,X
,C261 E8        INX
,C262 4C 57 C2  JMP $C257
,C265 EA        NOP
,C266 CA        DEX
,C267 E4 60     CPX $60
,C269 F0 0C     BEQ $C277
,C26B B0 0A     BCS $C277
,C26D A9 20     LDA #$20
,C26F EA        NOP
,C270 E8        INX
,C271 95 59     STA $59,X
,C273 E4 60     CPX $60
,C275 D0 F8     BNE $C26F
,C277 EA        NOP
,C278 86 53     STX $53
,C27A A0 00     LDY #$00
,C27C 4C 54 C0  JMP $C054
,C27F EA        NOP
,C280 18        CLC
,C281 A5 66     LDA $66
,C283 85 65     STA $65
,C285 A4 67     LDY $67
,C287 A2 00     LDX #$00
,C289 EA        NOP
,C28A B5 59     LDA $59,X
,C28C 91 64     STA ($64),Y
,C28E E4 60     CPX $60
,C290 F0 0A     BEQ $C29C
,C292 E8        INX
,C293 20 1C C2  JSR $C21C
,C296 4C 89 C2  JMP $C289
,C299 4C 39 C2  JMP $C239
```



```
,C29C EA       NOP
,C29D A2 00    LDX #$00
,C29F 20 1C C2 JSR $C21C
,C2A2 B1 64    LDA ($64),Y
,C2A4 D5 55    CMP $55,X
,C2A6 F0 06    BEQ $C2AE
,C2A8 C9 00    CMP #$00
,C2AA D0 F0    BNE $C29C
,C2AC EA       NOP
,C2AD 60       RTS
,C2AE EA       NOP
,C2AF A5 65    LDA $65
,C2B1 85 66    STA $66
,C2B3 84 67    STY $67
,C2B5 EA       NOP
,C2B6 E8       INX
,C2B7 20 1C C2 JSR $C21C
,C2BA B1 64    LDA ($64),Y
,C2BC D5 55    CMP $55,X
,C2BE D0 DC    BNE $C29C
,C2C0 E4 60    CPX $60
,C2C2 F0 BB    BEQ $C27F
,C2C4 4C B6 C2 JMP $C2B6
,C2C7 4C 39 C2 JMP $C239
,C2CA A9 00    LDA #$00
,C2CC 85 FB    STA $FB
,C2CE A9 A0    LDA #$A0
,C2D0 85 FF    STA $FF
,C2D2 A9 9E    LDA #$9E
,C2D4 85 FE    STA $FE
,C2D6 A0 00    LDY #$00
,C2D8 A2 00    LDX #$00
,C2DA B1 FE    LDA ($FE),Y
,C2DC D5 55    CMP $55,X
,C2DE F0 23    BEQ $C303
,C2E0 C9 80    CMP #$80
,C2E2 B0 31    BCS $C315
,C2E4 C0 FF    CPY #$FF
,C2E6 D0 02    BNE $C2EA
,C2E8 E6 FF    INC $FF
,C2EA C8       INY
,C2EB B1 FE    LDA ($FE),Y
,C2ED C9 80    CMP #$80
,C2EF 90 F3    BCC $C2E4
,C2F1 C0 FF    CPY #$FF
,C2F3 D0 02    BNE $C2F7
,C2F5 E6 FF    INC $FF
,C2F7 C8       INY
,C2F8 E6 FB    INC $FB
,C2FA A5 FB    LDA $FB
,C2FC C9 FF    CMP #$FF
,C2FE F0 C7    BEQ $C2C7
,C300 4C D8 C2 JMP $C2D8
,C303 C0 FF    CPY #$FF
,C305 D0 02    BNE $C309
,C307 E6 FF    INC $FF
,C309 C8       INY
,C30A E8       INX
,C30B B1 FE    LDA ($FE),Y
,C30D D5 55    CMP $55,X
,C30F F0 F2    BEQ $C303
,C311 C9 80    CMP #$80
,C313 90 CF    BCC $C2E4
,C315 E9 80    SBC #$80
,C317 D5 55    CMP $55,X
,C319 D0 D6    BNE $C2F1
,C31B A5 FB    LDA $FB
,C31D 18       CLC
,C31E 69 80    ADC #$80
,C320 85 FB    STA $FB
,C322 A2 00    LDX #$00
,C324 95 55    STA $55,X
,C326 A9 00    LDA #$00
,C328 4C 52 C0 JMP $C052
```

LOS MEJORES DE INPUT COMMODORE

PUESTO	TITULO	PORCENTAJE
1.°	Commando	18,2 %
2.°	Green Beret	15,5 %
3.°	Skyfox	12,3 %
4.°	Kung fu master	10,2 %
5.°	Rambo	9,9 %
6.°	Uridium	9,6 %
7.°	Hardball	9,3 %
8.°	The Eidolon	5,3 %
9.°	The way of the tiger	4,9 %
10.°	Misión Imposible	4,8 %
		100 %

Para la confección de esta relación únicamente se han tenido en cuenta las votaciones enviadas por nuestros lectores de acuerdo con la sección «Lo Mejores de Input».



LAS SERIES TELEVISIVAS MAS EMOCIONANTES
TRASLADAN TODA SU ACCION A TU ORDENADOR

¡¡CONVIERTETE EN SU PROTAGONISTA!!

DISTRIBUIDOR EXCLUSIVO PARA ESPAÑA: ERBE SOFTWARE. C/. STA. ENGRACIA, 17
28010 MADRID. TEL. (91) 447 34 10 - DELEGACION BARCELONA, AVDA. MISTRAL, N.º 10 - TEL. (93) 432 07

KIT DE UTILIDADES VORPAL

El Vorpal Utility Kit es un conjunto de herramientas de software destinadas a la unidad de diskettes 1541, desarrollado por Epyx y reciéntemente distribuido en nuestro pais. Una carpeta incluye el disco y un manual de instruciones en castellano.

El Vorpal presenta un menú de 5 opciones :

(a)-Volver al BASIC
(b)-Utilidades para ficheros
(c)-Alineamiento de la cabeza y control de la velocidad de giro del motor
(d)-Utilidades Vorpal
(e)-Instalación de cargadores.

En términos generales se puede decir que el invento de Epyx es un multiplicador de velocidad.

Realmente,el incremento en la velocidad de transferencia de los datos entre ordenador y unidad de disco se basa en un cambio del sistema operativo del manejo de discos, realizado mediante software. Se altera es el contenido del kernal de la unidad al tiempo que se define un nuevo tipo de fichero, llamado Vorpal.

Cuando se lee y visualiza el directorio del disco aparecen a la derecha de los nombres de los ficheros tres letras, que indican el tipo de que se trata: PRG= Programa, SEQ= Secuencial, USR= Usuario, REL= Relativo. Bueno, pues el Vorpal produce una nueva definición: SR?.

La segunda opción del menú permite elegir de entre un amplio surtido de posibilidades tales como rename disk, validate, format (formateo en 25 segundos), delete,.etc....típicos del DOS (disk operating system) wedge. Sin embargo otras utilidades vienen a enriquecer los recursos del usuario de disco. Pack es un programa de copia rápida que permite rescartar los





ficheros Vorpal en otro disco a una velocidad respetable. Change file type convierte los ficheros de cualquiera de los 5 tipos en cualquiera de los otros 4; esto es de PRG a SR?, SEQ a PRG, etc.. Todas las combinaciones son posibles aunque la verdadera ventaja consiste en pasar los ficheros PRG a SR?.

Otra de las opciones atípicas es protect file y su antidoto unprotect. Son comandos que protegen (hacen imborrable) o desprotegen (vuelven a convertir en borrable) cualquier tipo de fichero incluido el SR?. Esta utilidad es bastante reducida en la práctica ya que uno no anda borrando ficheros, aunque es interesante poder desproteger alguno directamente.

La opción set boot file permite que el Vorpal reorganice el BAM colocando el fichero en primer lugar en el directorio , de tal manera que luego baste con escribir LOAD"*",8,1 para cargarlo.

La tercera opción incluye un ali··ador de cabezas. Se supone que el ajuste se realiza automáticamente sin tener que manipular partes mecánicas. El programa monitoriza la alineación mediante un número. Si esta comprendido entre 1 y 25 la alineación es buena, entre 26 y 50 es aceptable y 51 o mas indica desastre. No tuvimos oportunidad de comprobar su funcionamiento, pues nuestras unidades de disco indicaban en todo momento un 26.

De un modo igualmente sencillo se puede testear la velocidad de giro del disco. Una cifra indica el funcionamiento de la unidad. Si está comprendido entre 297 y 303 es correcto, en caso contrario habrá que acudir a un servicio técnico. Nuestras

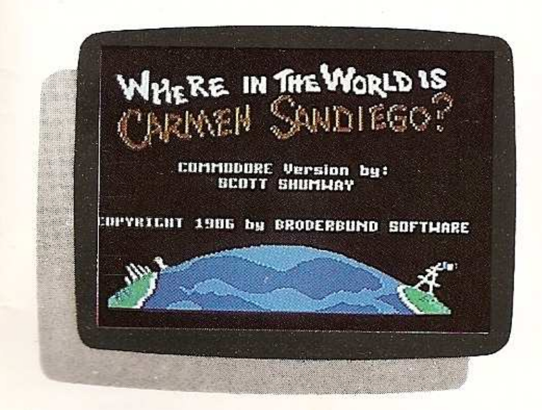
unidades arrojaron siempre un 299 y dos decimales variables.
La cuarta opción es un copiador de discos no protegidos. Es bastante bueno y sobre todo útil.
Por otro lado el cambio de sistema operativo del disco altera el modo de almacenar la información. El Vorpal genera bloques contiguos bisecuenciales, de este modo se graba pista tras pista. Este tipo de almacenamiento no es clásico en otros sistemas operativos.
El manejo de este kit es tremendamente fácil, realizandose integramente por menús. Lo mas notable es el importante aumento de la velocidad de trabajo con que se dota a las hasta 5 unidades de disco que pueden controlarse con el Vorpal.

****

SOFTACTUALIDAD

¿DONDE ESTA CARMEN SANDIEGO?

Un programa con un nombre tan curioso no es otra cosa que un juego conversacional, en el que debes atrapar al jefe de una organización internacional de ladrones profesionales cuyo cerebro parece ser la senorita Carmen Sandiego.
El nombre clave que te asignan para la misión Rockie y eres un agente de la Interpol.
De ahora en adelante pasaras un tiempo en los aviones que te llevan a visitar a las islas Seychelles e incluso a un monje budista, al que han robado su toga, o a un tal Silvester Stulune en el Nepal, quien rueda una pelicula de boxeadores y le han robado una bobina.
La resolución de los casos te permitirá ir construyendo un perfil del jefe del clan.
El juego utiliza ventanas multiples en las que ves aparecer informes, notas de viaje, pistas, etc..
Las ventanas se superponen y desaparecen en la medida en que van necesitandose para dar mas realidad a la trama.





****

ROCK AND ROLL MAMA!!

Sueñas con emular a Jaeger o Phill Colins y estás frustrado. Rock and Roll Mama te da la solución. Te pone en casa un auténtico estudio de grabación en el que se incluye un curso de bateria completo y un método de aprendizaje en la maestria de la tecla.



Podrás ensayar en la bateria unos cuantos compases y pedirle al técnico de la cabina de control que reproduzca lo que acabas de tocar o que cambie algún parametro del sonido,etc..
El programa genera ficheros directamente compilados que puedes incluir en tus juegos.

****

LAS SIRENAS TAMBIEN AMAN







DATOS GENERALES

TITULO Mermaid Madness

FABRICANTE Electric Dreams

ORDENADOR Commodore 64

TEMA DEL PROGRAMA
Rescate submarino

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	9
INTERES	7,5
GRAFICOS	8,5
COLOR	8,5
SONIDO	8
TOTAL	41,5



Las sirenas viven muchos años y normalmente no representan la edad que tienen. Este es el caso de Myrtle, una oronda y voluptuosa princesa del fondo de los mares que acaba de cumplir 112 primaveras. Sus escamas se vuelven mas verdes cuando ve al submarinista Gormless Gordon. Sin embargo no hay graves problemas que les separen, porque toma aspecto humano cuando pisa la tierra.
Su amado queda atrapado entre los restos de un transatlántico hundido y tu misión (la de Mirtle), consiste en recorrer las cuevas en busca de elementos que ayuden a socorrer al infortunado buceador. Es una carrera contra el tiempo, pues las botellas de oxígeno tienen una capacidad limitada. El control se realiza directamente a través del joystick. La sirena debe evitar a la criaturas marinas, que le robarán energía al mas mínimo roce. Una botella inclinada representa la cantidad de energía vital que posee. El hallazgo de nuevas botellas ocultas aportará una cantidad de energía.
Para conocer cuán cerca se encuentran los amantes en cada momento, existe un corazón palpitante en la parte superior de la pantalla, que se acelera al aumentar la proximidad entre ambos.
Tanto los gráficos como la animación son excelentes. Es un juego entretenido, situado entre la aventura y el juego de arcade.

****

LA GUARIDA DEL DRAGON

Alguien poco acertado dijo que las aventuras con dragones y princesas eran cosas del perdidas en el pasado. Nada mas lejos de la relidad, como vamos a ver.
Este programa se remonta mucho tiempo atrás, en una epoca mágica, cuando un rey llamado Aethered reinaba en un pais gobernado por la paz y los buenos deseos. Lugar pleno de increíbles tesoros, pero el mas preciado por su valor y belleza era la propia hija del rey, la princesa Daphne.
Un mal dia triste y gris un malvado dragón exigió al rey y a sus subditos el abandono de sus tierras. Ante una rotunda negativa

raptó a la princesa, encerrandola en su palacio encantado, en el interior de una bola de cristal.
A partir de ahora ya tienes trabajo. Eres Dirk, el caballero entre los caballeros, y tienes solo un pensamiento: Penetrar en la guarida del dragón, destruirle y rescatar a Daphne.
El juego está dividido en 9 etapas, cada una de ellas proporciona un paso hacia el éxito de la empresa.
La primera es la caida del disco, una plataforma que actua como ascensor. Unos etereos genios malvados te soplarán con fuerza para obligarte a caer fuera del disco. Peligro, son 7 las malvadas y transparentes criaturas.
Después verás aparecer una planta a la derecha, pero no te bajes, puede ser mortal. Espera en el disco y aparecerá otro genio y la planta en la que tienes que bajar. No te entretengas, solo tienes unas décimas de segundo para saltar. En cuanto pises suelo firme puedes lanzar un respiro de alivio, porque verás como se estrella la plataforma en el fondo del abismo
Ya estás en la segunda fase: El vestibulo del craneo. Ahora debes atravesár un pasillo lleno de puertas desde las que eres agredido por murcielagos, huesos, craneos, esqueletos e incluso otras aves al final del pasillo. Observarás una caracteristica que aparece en otras dos fases posteriores (el cuarto de armas y la habitación del tentaculo, fases 4 y 6).
Los movimientos se aplican desde el joystick en el momento oportuno. Si lo haces a destiempo te cuesta una vida y solo tienes 5 vidas. Hay una pista, el programa te da un tono grave cuando haces un intento erroneo; pero mucha paciencia, porque tendrás que jugar infinidad de veces.
Si logras pasar esta fase, llegas a la cuerda que se quema. Tras saltar sobre una superficie de piedra que se va desmoronando, tendrás que ir de cuerda en cuerda saltando como Tarzan, con la agravante de que las cuerdas se van quemando.
El final del juego es evidente : Una lucha a muerte con el dragón.



DATOS GENERALES

TITULO Dragon's Lair

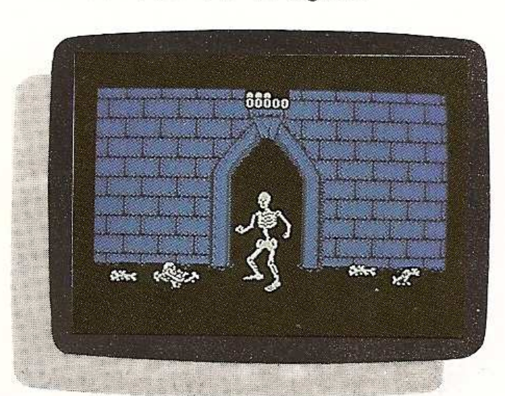
FABRICANTE Software Dreams

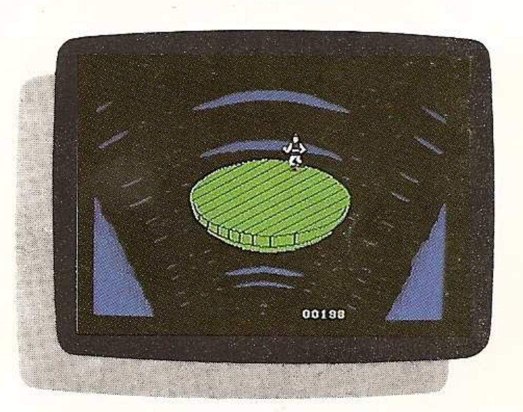
ORDENADOR Commodore 64

TEMA DEL PROGRAMA
Aventura medieval

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	8,5
INTERES	9
GRAFICOS	9
COLOR	8
SONIDO	7,5
TOTAL	**42**





Dragon's Lair está basado en el juego del mismo titulo que emplea una combinación de dibujos animados en videodisco (leido con ayuda de laser) y videojuego. A pesar de las enormes diferencias que le separan del original, esta versión para microordenador es de una gran altura, tanto por la cuidada calidad de sus gráficos como por el sonido. El único reproche que le hacemos es la lentitud en reiniciar el juego cada vez que se pierde una vida.

CORRUPCION EN MIAMI

Tu ordenador está a punto de convertirse en el santuario de la corrupción que en Miami hay. Sin embargo, Crockett y Tubbs están allí para limpiar sus circuitos de cualquier atisbo de crimen organizado.



Mientras se carga el programa aparece un revolver humeante junto al logotipo de la popular serie televisiva.
Por lo pronto puedes pilotar- via joystick- el fastuoso Ferrari-Testarosa de Sonny, comenzando a patrullar por las calles de la ciudad entre las verdes palmeras de Florida. Te acompana la pegadiza melodía musical de la serie. El control de un coche tan potente se hace difícil y puedes estrellarlo si no conduces con precaución. Una vista aerea del entorno del automóvil va desplazandose en la medida en que avanzas.
Recibes un soplo en torno a un alijo de contrabando a punto de llegar y has de reunír las pruebas y evidencias que te lleven a solucionar el caso. Has de entrevistarte con tus contactos habituales en determinados bares y establecimientos de la localidad.
LLegados a un punto de interés, ha de apearse uno de los dos (quien elijas), mientras el otro permanece sentado al volante. En ese momento el aspecto de la pantalla cambia, mostrando el interior del establecimiento, asi como al detective portando su Magnum en la mano. Aquí puedes encontarte con algún sujeto poco recomendable e incluso recogerás la heroina simplemente pasando sobre ella.
A su vez, la pantalla se divide en ventanas, apareciendo un reloj calendario que avanza inexorablemente y un indicador da el chivatazo cuando pasas ante algo que merece la pena investigar. Dispones únicamente de una semana para arrestar al peligroso mister J.
Asimismo aparecen las puntuaciones que vas logrando. Por otro lado se indica el modo en el que estás inmerso, sea busqueda, conducción... o presto a disparar. En todo caso siempre puedes elegír lo que hacer, en función del letrero parpadeante. No puedes realizar dos cosas al mismo tiempo, por lo que habrás de elegír en todo momento lo mas adecuado.
Un consejo: fijate bien en el coche rojoy, sobre todo, cuida tu pellejo en los antros del vicio de Miami.
El progreso en eljuego es lento, pero las emociones ocultas, los sensacionales gráficos y la pulida música hacen que Miami Vice sea cualquier cosa menos tedioso.



DATOS GENERALES

TITULO Miami Vice

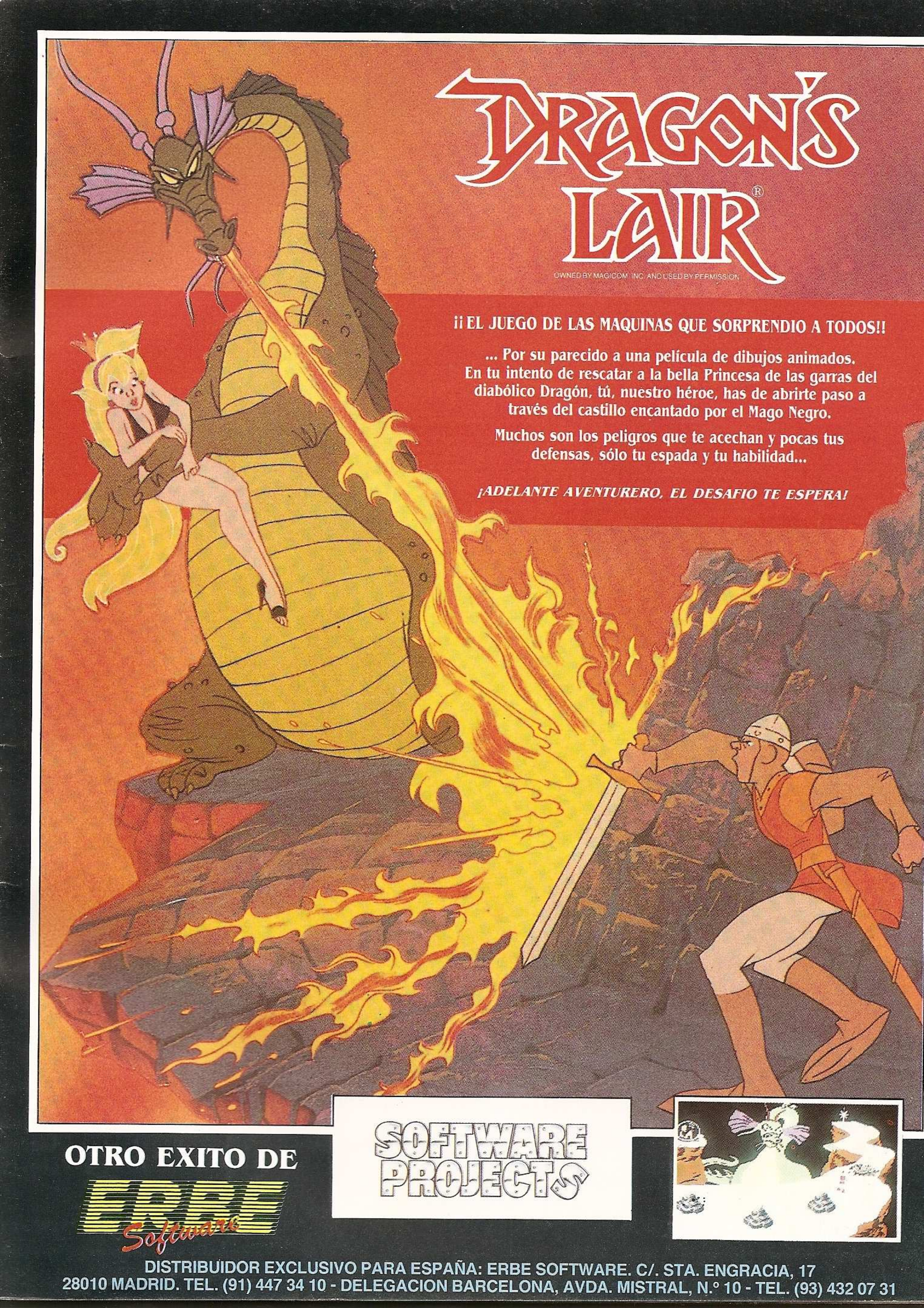
FABRICANTE Ocean

ORDENADOR Commodore 64

TEMA DEL PROGRAMA
Investigación policial

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	8,5
INTERES	9
GRAFICOS	8,5
COLOR	8,5
SONIDO	8
TOTAL	**42,5**


DRAGON'S LAIR®
OWNED BY MAGICOM INC. AND USED BY PERMISSION
¡¡EL JUEGO DE LAS MAQUINAS QUE SORPRENDIO A TODOS!!
... Por su parecido a una película de dibujos animados. En tu intento de rescatar a la bella Princesa de las garras del diabólico Dragón, tú, nuestro héroe, has de abrirte paso a través del castillo encantado por el Mago Negro.
Muchos son los peligros que te acechan y pocas tus defensas, sólo tu espada y tu habilidad...
¡ADELANTE AVENTURERO, EL DESAFIO TE ESPERA!
OTRO EXITO DE
ERBE Software
SOFTWARE PROJECTS
DISTRIBUIDOR EXCLUSIVO PARA ESPAÑA: ERBE SOFTWARE. C/. STA. ENGRACIA, 17
28010 MADRID. TEL. (91) 447 34 10 - DELEGACION BARCELONA, AVDA. MISTRAL, N.º 10 - TEL. (93) 432 07 31

DESCUBRE AL ASESINO

Muchos abandonaron en el camino, pero otros (pocos) llegasteis hasta el final en el reto planteado, Finalmente hubo que realizar un sorteo, imparcial por supuesto, al que asistimos elementos de la redacción de INPUT y personal de Anaya Multimedia.
Los mensajes planteados tenían como respuesta lo siguiente:



"ES UNA VIOLENCIA FINGIDA PUES SE TRATA DE UN TIPO DE CAJA INACCESIBLE A CUALQUIER TIPO DE PALANCA".

" LAS ATADURAS NO SON DE UN PROFESIONAL DA LA IMPRESION DE QUE TENIAN MIEDO DE LESIONARLE".

"TIENE LA COSTUMBRE DE MORDER LAS COLILLAS DE LOS CIGARROS QUE FUMA SIN PARAR HASTA DESHACERLAS".

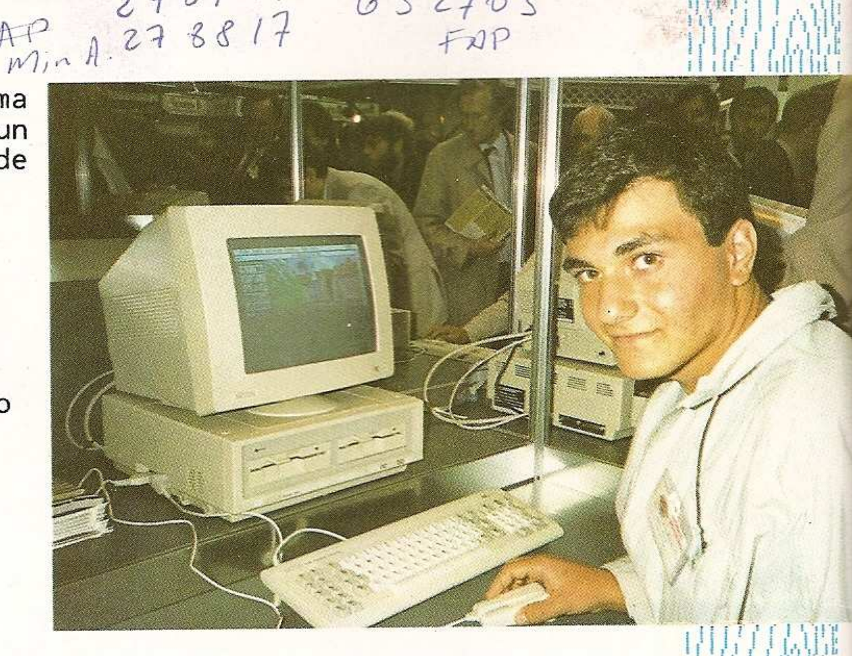
"EN EL REGISTRO EN LA CASA DE LA Victima aparecieron las joyas habilmente CAMUFLADAS BAJO EL SUELO DEL DORMITORIO".

"A PESAR DE ESTAR BASTANTE QUEMADA EN LA COLILLA SE PODIAN DISTINGUIR PERFECTAMENTE LAS MARCAS DEJADAS POR LOS INCISIVOS DE UN FUMADOR NERVIOSO".

278749 632703
FAP Min A. 278817 FAP

El intrépido y sagaz descifrador de claves se llama Manuel Bautista López. Todo un hallazgo para los servicios de inteligencia de cualquier pais, que no dudamos se lo rifarán cuando este ejemplar caiga en sus manos. Todo un criptofuturo por delante.

En la fotografía podemos apreciar a nuestro afortunado Manuel disfutando de su estancia en el PCW Show de Londres. Le acompana Santos Rodriguez, director de producto y delegado de Anaya Multimedia para el significativo evento.



Me puedes regalar...

una raqueta, un ajedrez,
un jersey, una estilográfica,
o una radio para el coche...

pero lo que yo necesito...
es este Commodore 128.



Y lo necesito porque es el más 128 de todo el mercado. Porque es 3 ordenadores en uno solo, sólo con pulsar una tecla.

Lo necesito porque es compatible con el C-64, y todos sus programas.

Lo necesito, porque ya que me decido, quiero lo mejor.

Lo necesito, porque todavía hay un par de vídeo juegos en los que me gana mi vecino y eso no puede ser.

commodore



Microelectrónica y Control, s. a. *Unico representante de Commodore en España. c/ Valencia, 49-53 08015 Barcelona - c/ Ardemans, 8 28028 Madrid*